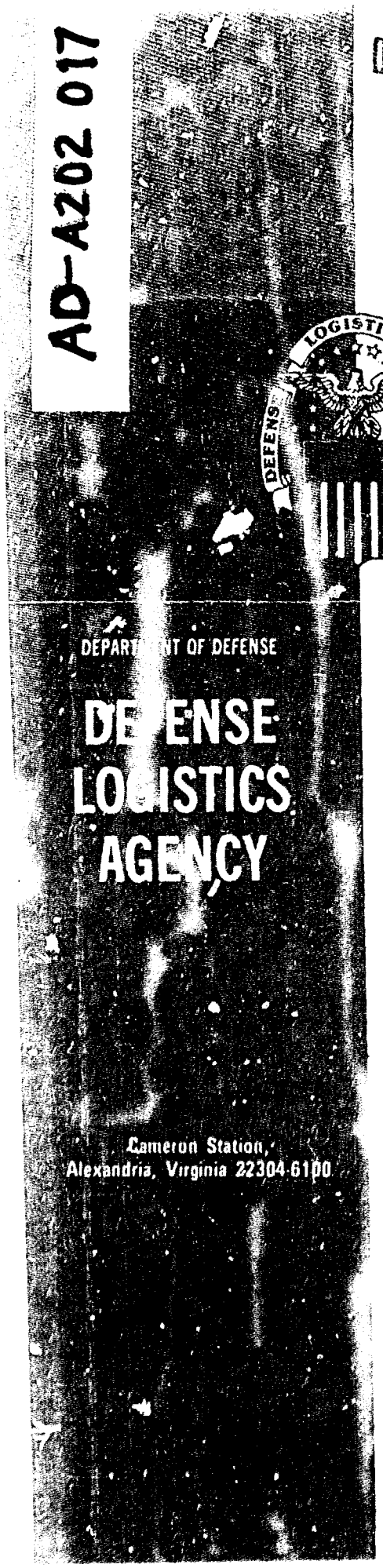


ANALYSIS OF QUALITY ASSURANCE (QA) EFFECTIVENESS

Operations Research and Economic Analysis Office

July 1988

Analysis of Quality Assurance (QA) Effectiveness

July 1988

Mr. Paul E. Grover
Operations Research and Economic Analysis Office
Headquarters Defense Logistics Agency
Cameron Station, Alexandria, Virginia 22304-6100

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION / AVAILABILITY OF REPORT Public Release; Unlimited Distribution
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Operations Research and Economic Analysis Office 6b. OFFICE SYMBOL (*If applicable*) DLA-LO	7a. NAME OF MONITORING ORGANIZATION Defense Logistics Agency (DLA-L)
6c. ADDRESS (*City, State, and ZIP Code*) Cameron Station Alexandria, VA 22304-6100	7b. ADDRESS (*City, State, and ZIP Code*) Cameron Station Alexandria, VA 22304-6100
8a. NAME OF FUNDING / SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (*If applicable*) DLA-L	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (*City, State, and ZIP Code*) Cameron Station Alexandria, VA 22304-6100	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.
11. TITLE (*Include Security Classification*) (U) Analysis of Quality Assurance (QA) Effectiveness	
12. PERSONAL AUTHOR(S) Paul E. Grover	
13a. TYPE OF REPORT Final 13b. TIME COVERED FROM May 86 TO May 88	14. DATE OF REPORT (*Year, Month, Day*) July 1988 15. PAGE COUNT 96
16. SUPPLEMENTARY NOTATION	
17. COSATI CODES FIELD / GROUP / SUB-GROUP	18. SUBJECT TERMS (*Continue on reverse if necessary and identify by block number*) effectiveness, quality assurance, productivity

19. ABSTRACT (*Continue on reverse if necessary and identify by block number*)

The lack of meaningful measures of effectiveness for the Quality Assurance (QA) function within the Defense Contract Administrative Services (DCAS) has been a serious deficiency for many years. With the availability of new, automated data from the QA Management Information System, these measures are made possible through the QUality Effectiveness Sensing Technique (QUEST) model which is documented in this report. QUEST evaluates the Government-driven Contract QA Program and the contractor-driven product conformance through a set of indicators using multi-attribute decision-making methods. These techniques combine quantitative statistical analysis with subjective factors provided by QA experts. The model was successfully tested by comparing QUEST measures with supervisors' opinions throughout DCAS organizational elements and is in test at one DCAS region.

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL Col Eugene Round, USAF	22b. TELEPHONE (*Include Area Code*) (202)274-6715 22c. OFFICE SYMBOL DLA-LO

DD FORM 1473, 84 MAR 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

DEFENSE LOGISTICS AGENCY
HEADQUARTERS
CAMERON STATION
ALEXANDRIA, VIRGINIA 22304-6100

DLA-LO
July 1988

FOREWORD

The lack of meaningful measures of effectiveness for the Quality Assurance (QA) function within the Defense Contract Administrative Services (DCAS) has been a serious deficiency for many years. With the availability of new, automated data from the QA Management Information System, these measures are made possible through the QUality Effectiveness Sensing Technique (QUEST) model which is documented in this report.

QUEST evaluates both the Government-driven Contract QA Program and the contractor-driven product conformance through a set of indicators using multi-attribute decision-making methods. These techniques combine quantitative statistical analysis with subjective factors provided by QA experts. The model was successfully tested by comparing QUEST measures with supervisor's opinions throughout DCAS organizational elements. It is, therefore, recommended that QUEST measures be implemented, as planned by the panel of experts that contributed to this model.

ROGER C. ROY
Assistant Director
Policy and Plans

CONTENTS

CONTENTS (Cont'd)

LIST OF TABLES

EXECUTIVE SUMMARY

Defense Contract Administrative Services (DCAS) organizations have used measures of efficiency for several years to determine the productivity of its Quality Assurance (QA) functions. However, DCAS has lacked uniform measures of effectiveness (MOE) concerning the degree of mission accomplishment for QA. With the implementation of a new automated data processing system, the Mechanization of Contract Administration Services, Phase II, data now exists which can produce such measures. The objectives of this study effort were to define QA effectiveness, to identify detailed MOE, and to develop a model which produces both detailed and aggregated MOE.

With the assistance of DCAS personnel and Study Advisory Group members, the DLA Operations Research and Economic Analysis Office has developed the QUality Effectiveness Sensing Technique (QUEST) model. QUEST measures QA effectiveness based on a dual definition of effectiveness. QA is effective when the Contract Quality Assurance Program (CQAP) is operated in accordance with established policies and procedures and the product is produced in accordance with contract provisions.

To measure program and product effectiveness, facilities (contractors) are peer grouped based on commodity, QA provision, and size. Facility data are compared against peer group averages and standard deviations to produce a relative measure. QUEST measures, therefore, represent how well the government QA representative (QAR) and the contractor together compare with others in similar circumstances.

Program effectiveness measures are primarily a function of the actions of the QAR. QUEST evaluates 17 CQAP functions and identifies "red flags" when an out-of-tolerance condition is detected. Flags can occur when the QAR fails to perform expected actions or when the actions performed are abnormal in relation to peers. QUEST counts flags and scores program effectiveness based on the number of flags generated per facility per month.

Product effectiveness is measured using a multi-attribute decision-making technique called Technique for Order Preference by Similarity to Ideal Solution (TOPSIS). This technique has been successfully used by the Army for officer promotion boards, by DLA to determine which items require Industrial Preparedness Planning and in many other applications. Its unique concept of defining best (ideal) and worst (negative ideal) scenarios as a reference for measurement is less subjective than many other multi-attribute techniques. Seven product-related indicators and their trends are measured individually and as a group to achieve a product MOE. These product indicators are primarily driven by the contractor's performance but can be influenced by the QAR.

QUEST provides an accurate measure of QA effectiveness, as demonstrated by the successful validation achieved by comparing model results with expert opinion obtained by questionnaires. Therefore, it is recommended that the QUEST model be implemented by the DLA Directorate of Quality Assurance in accordance with a plan developed by the QUEST Study Advisory Group.

I. INTRODUCTION

A. Background

The Defense Contract Administrative Services (DCAS) administers a wide range of contracts for the Military Services, Defense agencies and other Federal agencies. In-plant quality assurance (QA) responsibilities are implemented by DCAS through the Contract Quality Assurance (CQA) program (CQAP). This five part program is specified in DLAM 8200.1, Defense In-Plant Quality Assurance Program, and DLAM 8200.2, Procurement Quality Assurance Support Manual for Defense Contract Administrative Services.

The QA Management Information System (QAMIS) enables DCAS to manage and control certain aspects of the CQA program. Through data reported by individual QA representatives (QARs), information relating to contracts, contractors and QAR actions is systematically collected, reported and summarized by QAMIS. Prior to 1986 all regions utilized a batch process mode of QAMIS whereby the QAR-generated forms were mailed to automated data processing (ADP) operations and monthly hard copy reports were generated. Beginning at DCAS region (DCASR) Atlanta (DCASR ATL), an improved version of QAMIS was established as part of the Mechanization of Contract Administrative Service (MOCAS), Phase II Segment IX. In addition to on-line data input, on-line query capability, and other operational improvements, more detailed data elements are reported, providing better visibility of CQA operations. This new system is now operational at DCASR-Chicago (DCASR CHI), DCASR-Cleveland (DCASR CLE), DCASR-Dallas (DCASR DAL) and DCASR-St. Louis (DCASR STL). Plans are underway to extend the new QAMIS to the other regions.

As a result of the new data elements available under the new QAMIS, the DLA Operations Research and Economic Analysis Office (DLA-LO) was tasked to develop a model that produces measures of effectiveness of the DCAS CQA program.

B. Problem Statement. To develop a methodology to measure the relative effectiveness of CQA operations from the contractor facility level to the regional level.

C. Objectives

1. To define CQA "effectiveness."

2. To define measures of CQA effectiveness that are compatible with available data.

3. To develop analytical methodologies to compute measures of effectiveness.

4. To develop a methodology to combine individual measures of effectiveness into a consistent measure of overall relative effectiveness.

D. Scope

1. Measures of effectiveness are limited to using those data elements reported under QAMIS, Phase II. No new or additional reporting burdens will be generated to support this effort.

2. Model development is limited to data bases established at the five Phase II DCAS regions beginning in January 1986 through July 1987.

E. Approach. Extensive use was made of experienced supervisory and staff experts at the DCASRs and DLA Directorate of Quality Assurance. Brainstorming sessions with DCASR personnel produced the input resulting in accomplishing the first two objectives specified in paragraph IC. A Study Advisory Group (SAG), shown in Appendix A, was formed to screen and augment the input from the brainstorming sessions, to provide subjective input such as weighting factors and to guide and assist the analysts in conceptual development. The SAG also assisted in the validation of the model and in developing a plan to implement the model. The modelling approach was designed to emulate the thought processes of knowledgeable experts as they subjectively and objectively view the effectiveness of QA.

II. METHODOLOGY. QUEST is a FORTRAN based model. Its operation is designed for an IBM operating environment, using certain IBM utility programs and IBM JCL. Generally, QUEST extracts relevant data from QAMIS and processes the information systematically to produce both detailed and summary level measures of effectiveness. This is done by first assigning each facility to a peer group. Then monthly QAMIS data for that facility are compared to its historical peer group averages to assess the relative effectiveness of the CQA program and the potential for product non-conformance.

A. Data Sources. Raw data to support this effort are needed from four sources. Data records for the QUEST model are developed by merging data elements from these sources by matching the Commercial and Government Entity (CAGE) codes (formerly Federal Supply Codes for Manufacturers (FSCM)). The term FSCM will be used throughout the remainder of this report to refer to facilities.

1. Performance History File. Each month a Phase II DCASR produces a tape containing records for each FSCM reported under QAMIS. The reported elements are provided in Appendix B and defined in DLAM 8200.2. Generally, these data provide hours, counts and statistics pertaining to QA activities during that month.

2. Facility Profile. For each FSCM under CQAP, the QAR enters descriptive data about that facility when that facility is first monitored. As conditions change, the QAR updates the Facility Profile. Each month a Phase II DCASR produces a Facility Profile Tape as shown in Appendix B.

3. Materiel Deficiency Report (MDR) File. Each Phase II DCASR maintains an on-line file for monitoring Materiel Deficiency Reports. These reports and available data elements are described in DLAR 8260.2, Materiel Deficiency Investigation and Reporting System. Because of the file structure of the on-line MDR file, a special program is required to produce a "flat" file of fixed length records needed by QUEST. This utility based program must be run each month prior to running the QUEST model to produce a file shown in Appendix B.

4. Contractor Alert File. Each month DCASR Boston issues a list of contractors that have a history of past problems. The purpose of the alert file is to warn acquisition elements that a DCASR has experienced problems and to provide data concerning the nature of these problems. Approximately 1000 FSCMs currently appear on the alert list nationwide. An "alert" can be issued because of quality, production, technical or financial problems.

B. Data Base Establishment. Prior to execution of the QUEST model, a data base must be established and maintained as follows.

1. MDR Processor. The purpose of this module is to identify "valid" MDRs, assign weighting factors to reflect the "age" of the MDR and to prepare the MDR record for merging with the history file. This record contains, for each facility, the weighted number of valid MDRs and the average days to close an MDR per month. See Appendix C for additional details.

2. File Merger. A module documented in Appendix D is used to create a historical data base of up to 120 months of data on tape. The input files must be sorted by FSCM by year by month in ascending order prior to this step. Because of the dynamic nature of the MDR file (also to a lesser degree the performance history file and the facility profile), it will be necessary to not only merge the current month into the data base, but also to recreate several past months each month. Appended to the performance history file are the following:

a. Number of government QA personnel assigned to the facility from the Facility Profile.

b. Contractor Operation Type Codes (i.e., manufacturer, overhaul/repair, distributor/jobber, etc.) from the Facility Profile.

c. Weighted MDR count from the MDR processor.

d. Average days to close MDRs from the MDR processor.

C. Data Selection. This section describes how the QUEST model selects data for processing from the tape data file described in the previous section. A screening process is detailed in Appendix E.

1. Program Control. A small input file, documented in Appendix E, must be created to control the execution of the QUEST model. This file specifies parameters used in the screening process.

2. Record Selection. Data records from the History Tape are screened to determine if further processing is in order. These selected records are stripped of extraneous data elements and put into a disk file. Reasons for non-selection are inactivity, invalid or missing data and dates outside the horizons specified for model execution. Appendix E further describes this process.

D. Stratification Process. To compute effectiveness measures QUEST interprets QAMIS data by comparing raw data against average values for similar facilities. The deviation from average translates into a relative effectiveness score. QUEST measures of effectiveness are, in reality, measures of facilities relative to a peer grouping established by QUEST model logic based on facility size, dominant commodity and QA provision. QUEST assigns a peer group number to each facility evaluated. This grouping process is described in general below. Appendix F provides details on grouping methodology and additional documentation is shown in Appendix E, pages E-9 through E-12.

1. Grouping Process. Criteria for grouping facilities was constrained by the need to have a sufficient number of facilities to compute reliable averages and standard deviations for QAMIS data elements. Thus grouping was accomplished in two steps. Using a theoretical, unconstrained logic the facilities were grouped. Based on the unconstrained grouping, a statistical analysis was run to count the number of observations in each group. When the number of observations was small, logical groupings of "low hit" groups were sought to combine into a larger group.

2. Grouping Logic. A theoretical logic to group facilities was designed and approved by the SAG. However, it became necessary to depart from theory because of the extreme lack of homogeneity of groups. Some groups have hundreds of "peers;" others have very few or none. A two step process used to overcome this problem is described as follows:

a. Unconstrained Groups

(1) Resident Facilities. Because of the limited number of Resident facilities, only 240 groupings (16x3x5) were attempted based on:

(a) Commodity Code. The first alpha commodity code is selected. Commodity code is a two alpha code where the first alpha is the primary commodity designator and the second alpha further defines the commodity within the primary grouping. Commodity Codes are defined in DLAM 8200.2. There are 16 primary commodity codes.

(b) Quality Provision. Grouping by QA Provision is necessary because of significant differences in QA activity caused by the provision. Three provision codes are used for MIL-Q-9858A (MILQ), MIL-I-45208A (MILI) and standard inspection provisions.

(c) Facility Size. Size is measured by the number of Government QA personnel assigned to the facility. Five size groupings were selected as follows:

1 0 - 2 QA Specialists (QAS)

2 3 - 7 QAS

3 8 - 20 QAS

4 Reserved for future use

5 Over 20 QAS

(2) Nonresident Facilities. Because there are more nonresident facilities than resident facilities, more detailed groupings were feasible. There were 384 groups (16x8x3) established for nonresident facilities.

(a) Commodity Code. The two alpha commodity code was used. There are 16 primary codes in the first alpha position and up to 8 in the second position.

(b) QA Provision. Three possible codes are available to describe the QA provision (MILQ, MILI or standard inspection).

(3) Maintenance Facilities. Because of the unique QA conditions that exist in maintenance facilities versus production facilities, three separate groups were set up for maintenance facilities. Maintenance facilities were defined as those with an Operation Type Code of "C" (overhaul/repair) or Commodity Code "A5" for Aircraft Programmed Depot Maintenance commodity. The three maintenance groups are non-resident, resident under eight government QASs and resident eight or more QASs.

(4) Any group not assigned to one of the previous groups (because of missing or unexpected codes) was also assigned a "catch-all" group number. Thus there are a total of 628 different potential groupings in the unconstrained case.

b. Constrained Groups. Despite the fact that there are several thousand facilities reported by the five Phase II regions, most of the facilities are clustered into a few popular groupings. Many of the unconstrained groupings have few or no facilities that meet the grouping criteria. Therefore, these groups were combined logically to ensure that each peer group has a reasonable number of "similar" facilities to establish group norms and standard deviations. Commodity codes were combined per logic shown in Appendix F.

c. When additional DCAS Regions operate under Phase II, and as more data is collected at the existing regions, it is expected that some of the inter and intra commodity combinations can be re-segregated.

3. Creation of a Master Data File. An integral part of the QUEST model is an external file that contains historical statistics on QAMIS data elements broken out by peer grouping. This file was developed by merging the Phase II regions' records described in paragraph IIC2. A standard statistical package was used to compute the average and standard deviation for key data elements by group identification number. Rules used to create the Master Data File are:

a. The first month of available data from each region was discarded because of warnings that the initial data was unreliable. Note: The master data file was created from data collected from February 1986 through April 1987. Because Phase II was phased into the regions, the number of observations was skewed towards regions that have been using Phase II longer. The DCASR-ATL contribution to the statistics is much greater then the DCASR-DAL contribution.

b. For groups that had no or few observations, a hypothetical record was created by substituting a similar group.

(1) Resident Facilities. It was observed that facility size was the dominant factor in most QAMIS data elements. Therefore a size group average (crossing commodity and provision boundaries) was computed and substituted for void or sparse groups.

(2) Nonresident. If a group was void or sparse, non-resident commodity averages were used for the Master Data File.

c. The Master Data File was reviewed by the Study Advisory Group. Rare inconsistencies were resolved by using size or commodity peculiar statistics in lieu of actual group statistics.

d. The Master Data File will require periodic update. With more facilities upon which the statistics are generated, and with shorter time gaps between the Master Data File and the QUEST measurement timeframe, QUEST comparisons become more valid.

E. Calculation of Measures of Effectiveness. Having grouped each facility by assigning a group number and having established what is normal for that group via the Master Data File, the next step for the QUEST model is to evaluate each facility for QA effectiveness. Appendix E contains further model documentation.

1. Definition of QA Effectiveness. QA at a facility is effective when, compared with its peer group, the facility demonstrates that both:

a. its operations conform to established CQA Program policies and procedures and;

b. through the use of CQAP, the product conforms to the contractual requirements.

2. Program Indicators. CQAP Effectiveness measures were designed to evaluate QAMIS data to determine if the data indicate that the QAR is following established CQAP regulations and guidance. The QUEST model reviews monthly data, seeking conditions that indicate required actions are not being taken or unusual out-of-tolerence conditions exist. When the model detects such conditions, an appropriate "Red Flag" is generated. The CQAP measure of effectiveness is driven by the number of "Red Flags" generated.

a. Red Flags. Based on brainstorming sessions with Phase II DCASR personnel and input from the Study Advisory Group, 17 conditions are tested. These conditions indicate a potential problem that may require follow-up by the QAR or the first line supervisor. Presence of a flag does not always mean that the QAR is not performing CQAP properly. For example, a flag may be caused by a data reporting error. Also, flags may occur because of special processes and conditions that exist that are peculiar to a given facility. For example, a facility that was flagged for shipping product without inspection turned out to be a coal mine which required inspections to be done at a lab (which was reported as another facility in the QAMIS). Therefore, the CQAP measure is only an indicator of potential effectiveness or ineffectiveness. Table 1 identifies the specific flags which are further described in Appendix G and documented in Appendix E-20 through E-23.

Table 1

RED FLAGS

A	Lots or Units Rejected without Corrective Action (QDR)
B	Intensified Inspection without Corrective Action
C	Abnormal Corrective Action Distribution
D	Corrective Actions without Meeting with Contractor
E	Shipments without Product Verification Inspection
F	Shipments without Visits - Nonresident
G	No Procedure Evaluation at MILI or MILQ Facility
H	Contracts Received without Planning
I	Lots Rejected without Reinspection
J	Excessive Days to Close a Materiel Deficiency Report
K	Excessive Net Quality Assurance Letters of Instruction
L	Non QAR QDR Actions
M	Excessive System Indicator
N	Excessive Principal Contracting Officer Requests
O	Excessive Contract Administrative Office Requests
P	Excessive Administrative Hours
Q	Work Not Performed

b. Flag Counts. In addition to identification of Red Flags, QUEST counts the number of flags generated per facility per month. It is assumed that all flags equally measure CQAP ineffectiveness. The model computes a score, called Program Effectiveness, based on the premise that no flags equals perfect Program Effectiveness and that the more flags generated the lower the Program Effectiveness. Thus a no flag facility is scored 100%, and points are deducted as flags are counted.

(1) It was observed that larger facilities tended to receive more flags than smaller facilities. Thus the penalties per flag were greater for small facilities than for large facilities to prevent bias.

(a) Nonresident Facilities. The penalty per flag is 20% per flag. If there are five or more flags in a given month, the Program Score is "clipped" at 0%.

(b) Resident Facilities (less than 20 QASs). The penalty per flag is 15% except for the first flag which carries a 10% penalty. This rationale for lower penalties per flag was based on observations that resident facilities were more prone to flags because more activity occurs. Thus there are more opportunities for problems. The 10% penalty for the first flag was justified by the high percentage of facilities that get a single flag. The SAG felt that a single flag should result in a reasonably good Program Effectiveness score. If 7 or more flags occurred, the score was clipped to 0%.

(c) Resident Facilities (more than 20 QASs). It was observed that very large facilities (DCAS Plant Representative Offices (DCASPRO)) tended to receive more flags then other resident facilities. To normalize the scores, the penalties were adjusted to be 15% per flag except for the first two flags which carry only a 10% penalty. The score was clipped to 0% for 8 or more flags.

(2) Table 2 summarizes the Penalty function used by QUEST to compute Program Effectiveness.

Table 2

RED FLAG PENALTY FUNCTION

Program Effectiveness %

Number of Flags	Nonresident	Resident less than 20	Resident over 20
0	100	100	100
1	80	90	90
2	60	75	80
3	40	60	65
4	20	45	50
5	0	30	35
6	0	15	20
7	0	0	5
8 +	0	0	0

3. Product Indicators. Unlike the program indicators discussed in the previous paragraph, product indicators based on counts or count ratios are continuous, monotonic, and negative in value. In other words, as the count or ratio increases for the identified product indicators, the implied quality of the end product is believed to be lower. Unlike red flags, then, which are either present or not present, product indicators can be quantified on a continuous scale and readily combined using multi-attribute techniques. The process of evaluation of product quality is explained in the following:

a. Identification of Product Indicators. Seven indicators were identified by experts during brainstorming sessions and SAG meetings. Experts indicated that not only is the magnitude of the indicator important in assessing product quality, but the rate of change or trend is equally important in measuring effectiveness. The indicators selected for use in measuring product quality effectiveness are shown in Table 3 and further discussed in Appendix H.

Table 3

PRODUCT INDICATORS

Estimated Process Average	EPA
Lot Rejection Ratio	LRR
Material Review Board	MRB
Waivers and Deviations	W/D
Engineering Change Proposals	ECP
Corrective Actions	CA
Materiel Deficiency Reports	MDR

b. Normalization of Indicator Values. QAMIS data was normalized on a scale of -3.0 to +3.0 representing (roughly) the number of standard deviations from average as outlined in Appendix H.

c. Calculation of Individual Indicator Scores

(1) Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) [2]. The TOPSIS technique developed by Dr. Ching-Lai Hwang of Kansas State University is a multiple attribute decision making technique. TOPSIS enables the analyst to deal with several variables that contribute to an overall objective by combining the multiple variables into a single factor in a logical and common-sense manner. Basically, TOPSIS works by:

(a) Defining a set of positive ideal and negative ideal conditions for the attributes involved. In other words, the best and worst possible cases are established.

(b) Measuring any set of attributes in terms of distances from both the positive ideal and negative ideal points. A ratio of distances is computed to produce a relative score. This ratio is the distance from a given point to the negative ideal point divided by the total distance from the point to both extremes.

(c) Therefore, if a point is far from the negative ideal and close to the ideal point, the ratio of distances is large and the score is high. On the other hand, if a point is near the negative ideal, the ratio of distances is small and the relative score is low.

(2) Definition of "Ideal" and "Negative Ideal" states. When experts were pressed to define an ideal set of conditions for the seven indicators, a common response was "It depends ....." The primary factor in determining whether or not a certain indicator value was "good" or "bad" seemed to be the nature of the contractor. The interpretation of QAMIS data from a contractor facility that has a history of quality excellence versus a contractor that has a poor track record is different. For example, a sudden upturn in corrective action activity for a historically good contractor could indicate that there are problems developing within the contractor's program or that the QAR is unfairly harassing the contractor. Both of these possibilities are negative and should result in an effectiveness degradation. On the other hand, the same data concerning a "problem" facility would indicate that the QAR is finding and correcting problems, probably resulting in improved product quality.

(a) The criteria to identify whether or not a contractor is a problem contractor are presence on the Alert list (see paragraph IIA4), high MDR counts and serious corrective actions. Specifics are provided in Appendix H.

(b) If the contractor is "normal." The definition for ideal and negative ideal conditions are the same for all seven indicators. Ideally all indicators are three or more standard deviations below average and decreasing at rates of 3 or more standard deviations (Z = -3 and -3 for indicator value and rate). Negative ideal is the opposite state (Z = +3 and +3).

(c) If the contractor is a "problem" contractor, it is the QAR's job to try to improve the contractor's quality system. Therefore the rate of change of the indicators tends to be dominant or more important than the indicator values themselves. In other words, it is already recognized that there are serious quality deficiencies in the facility, so the emphasis becomes to take action to get better. For EPA, LRR, MRB, W/D and ECP indicators, ideal is a rate decrease of three or more standard deviations and negative ideal is a 3 standard deviation or more rate increase. For Corrective Action (CA), the rationale explained in paragraph IIE3c(2) results in a definition of ideal CA to be +3 standard deviations above normal. For problem facilities, the QAR should be issuing many corrective actions, thus preventing product quality problems. Negative ideal CA definition is when CA counts are three standard deviations or more below average. With a known problem contractor, absence of sufficient QDR activity may indicate that the

QAR is failing to assert control. Material Deficiency Reports (MDR) are the most directly linked indicator to end product quality. As such, MDR counts and not rates are felt to more reflect the product quality. Even if the other indicators are decreasing, a continued high valid MDR count is viewed as a failure of the QAR and the contractor to ensure product compliance. Thus Z values of -3 standard deviations from average are ideal for MDR's for problem contractor. Z values of +3 are negative ideal for MDR counts at problem facilities. Appendix H contains additional information on ideal and negative ideal states.

d. Calculation of overall Product Score. TOPSIS enables the analyst to mathematically combine the individual attributes, weighted by relative importance.

(1) Weight factors were developed by the SAG using SPAN [1] methodology. Weights for resident facilities chosen were EPA = .77; LRR = .77; MRB = .91; W/D = .72; ECP = .32; CA = 1.00; and MDR = .98. For nonresident facilities, the weights are EPA = .66; LRR = .98; MRB = .83; W/D = .64; ECP = .32; CA = 1.00; and MDR = .98.

(2) Overall product score is computed using the ratio of Euclidean weighted distances in 14 dimensional space (7 dimensions for problem facilities) as shown in Appendix H.

4. Total Scores. A final numerical score is computed by taking a weighted average of the program and product scores. The program score is weighted 60% and the product score was weighted 40%. Weights were provided by the SAG using SPAN methodology [1]. This total score is also called the QUEST score.

F. Report Generation. The final module of the QUEST model produces a hard copy printout detailing the monthly effectiveness scores produced by QUEST. After QUEST has computed the measures of effectiveness for all facilities in the target region during the months specified by the Program Control, an output file is created. This file is sorted by year, month, organization code, facility type, and QUEST score. This file is the data source for the report generator. The purpose of the report generator is to produce an informative, one page summary for each section and to compute average measures at various levels of the organization. Table 4 is a sample of the report. Appendix I contains further documentation.

1. Report Characteristics. The following information and corresponding values shown in Table 4 are reported.

a. Date. The month and year of the QAMIS data used to measure QUEST scores appear in the upper left corner; i.e. 7-87 representing July 1987 measurement.

Table 4

Sample Output

7 87 EFFECTIVENESS REPORT FOR SECTION A11

FSCM	GRP		RED FLAGS	PROGRAM SCORE	EPA	LRR	MRB	WVRS& DEVS	ECP	CA	MDR	PRODUCT SCORE	TOTAL SCORE	PRIOR MONTH	PEER RATING
90454	152	C	PQ	60.0	100.0	75.0	100.0	0.0	75.0	53.9	75.0	64.0	61.6	74.8	F
56221	2		I O	75.0	50.0	16.0	56.2	100.0	100.0	56.0	99.9	61.4	69.5	79.2	F
4H618	353		I O	60.0	56.4	63.3	75.0	75.0	75.0	36.0	75.0	61.1	60.4	68.3	F
21053	387		PQ	60.0	75.0	75.0	75.0	75.0	75.0	24.9	75.0	61.8	60.7	79.4	F
5X226	464		FG	60.0	75.0	75.0	75.0	50.0	75.0	50.0	75.0	66.3	62.5	62.5	F
1K459	601		E K	60.0	75.0	75.0	47.3	75.0	75.0	74.9	75.0	69.9	63.9	64.3	D
1Z621	464		G Q	60.0	75.0	75.0	75.0	75.0	75.0	100.0	75.0	78.6	67.4	79.4	D
1HS88	519	A		80.0	75.0	50.2	75.0	75.0	75.0	18.5	75.0	56.1	70.4	84.4	F
07666	458		O	80.0	50.0	75.0	75.0	50.0	50.0	21.9	75.0	56.9	70.7	64.1	C
6Y662	462		I	80.0	73.3	71.9	75.0	75.0	75.0	35.9	75.0	64.1	73.6	57.1	F
*3M710	458			100.0	50.0	50.0	50.0	50.0	50.0	0.0	100.0	39.3	75.7	63.7	C
*5A517	251			100.0	50.0	50.0	62.3	50.0	50.0	0.0	100.0	40.3	76.1	75.3	C
9A180	290	C		80.0	61.3	100.0	100.0	62.5	75.0	57.3	100.0	76.6	78.6	61.5	C
2X782	458		G	80.0	75.0	75.0	75.0	75.0	75.0	91.0	75.0	77.8	79.1	91.4	C
3L775	260			100.0	75.0	75.0	75.0	100.0	75.0	8.3	75.0	59.2	83.7	90.6	B
4U716	387			100.0	75.0	75.0	75.0	75.0	75.0	50.0	75.0	68.6	87.4	87.4	C
68424	525			100.0	75.0	75.0	75.0	75.0	75.0	62.4	75.0	71.8	88.8	88.8	B
27253	363			100.0	75.0	75.0	75.0	75.0	75.0	75.0	75.0	75.0	90.0	90.0	C
8L152	581			100.0	75.0	75.0	75.0	75.0	75.0	75.0	75.0	75.0	90.0	90.0	A
23364	464			100.0	75.0	75.0	75.0	75.0	75.0	75.0	75.0	75.0	90.0	91.4	A
63939	257			100.0	75.0	75.0	75.0	75.0	75.0	75.0	75.0	75.0	90.0	90.0	A
2U569	339			100.0	75.0	100.0	75.0	100.0	75.0	74.8	75.0	79.9	92.0	92.0	B
SUBTOTALS	A11			67.5	75.0	45.5	78.1	50.0	87.5	54.9	87.4	62.7	65.5	77.0	
RESIDENT				85.0	69.5	73.0	73.0	71.9	71.3	50.3	78.8	66.4	77.5	77.1	
NONRESIDENT				73.3	73.2	54.7	76.4	57.3	82.1	53.4	84.5	63.9	69.5	77.0	
COMBINED															

b. QA Organization Code. The three alpha QA-Org Code appears in the top line and in the subtotal line following the dashed line. This data element identifies the division (first alpha), branch (second alpha) and section (third alpha). For example All represents Division A (DCASMA Atlanta), Branch 1 (Atlanta Operations), Section 1 (Marietta) within DCASR Atlanta.

c. FSCM. Identification of facilities is by FSCM. FSCM's above the blank line are resident facilities. Inactive facilities are not reported. If there were no hours expended against an active facility in a particular month, no line will appear for that month. If an asterisk (*) appears before the FSCM, the facility was identified as a "problem" facility by QUEST as described in Appendix H. QUEST limits the number of active FSCM's per section to 500 per month.

d. GRP. The peer group number is shown for each FSCM. For example, FSCM 90454 was assigned to group 152, a resident, nuclear, MIL-Q facility with 3 to 7 government QAS's in residence per Appendix F.

e. Red Flags. Individual red flags identified are shown by letter. For example, in July 1987, FSCM 90454 received three flags (C, P and Q) resulting in the following Program Score.

f. Program Score. The number of flags generated produces a Program Score in accordance with Table 2. FSCM 90454, a resident facility with less than 20 QAS's, is reported as 60% effective on its program score (a very poor rating).

g. Individual Attribute Scores. Each of the seven product related indicators are scored using the TOPSIS methodology and reported. For example, FSCM 90454 scored very high concerning EPA and MRB but scored very poorly concerning waivers and deviations. Generally speaking, individual product scores over 70% are usually above average.

h. Product Score. A TOPSIS generated "weighted average" combining the individual product indicators scores is reported. The product score of 64% for FSCM 90454 is slightly below average.

i. Total Score. Combining the program score and the product score by a 60/40 ratio respectively produces the total score. Within facility type, the sequence of FSCM's reported is in ascending order of Total Score. The QUEST rating of 61.6% for FSCM 90454 is a relatively low rating, caused primarily by the low program score.

j. Prior Month. The previous QUEST rating produced in the prior month of record is shown. For example, FSCM 90454 scored a 74.8% Total Score in June 1987. The reason for the sudden drop in ratings from June to July was the appearance of the P flag (Admin Hours) and a very large number of waivers and deviations received in July.

k. Peer Rating. In addition to the ability to compare FSCM's within sections, the users desired the ability to evaluate a FSCM against its peers DLA-wide. To simplify the comparison, a letter grade of A, B, C, D or F is assigned based on a comparison of the Total Score with a Peer Group average Total Score. Since FSCM 09454 obtained a 61.6% Total Score and because Group 152 facilities DLA-wide average 84% with a standard deviation of 7%, this facility is 3.2 standard deviations below average. This parameter equates to the letter value "F." Each peer group has its own set of average and standard deviations. Some peer groups tend to be more "ineffective" than others on the average. Thus a score of 90% for FSCM 27253 (nonresident, general, lumber and wood products, standard inspection) rates only a "C" because historically, this group has a high average (86.6%) whereas a 90% for most groups would result in a "B" or "A" rating.

2. Sectional Subtotals. Averages by Facility type were computed for each section for all percentage scores discussed in the preceding paragraph.

a. Resident. The resident subtotal is the weighted average of resident facilities reported during the month of record at the section. The weight assigned is roughly proportional to the size of the facility measured by the number of government QASs. The relative weights assigned are 0 - 2 QASs = 1.0; 3-7 QASs = 4.0; 8-20 QASs = 9.0 and over 20 QASs = 25.0.

b. Nonresident. A simple average is computed for all scores for all nonresident facilities within the section.

c. Combined. The weighted overall average for all facilities is computed. Individual nonresident facilities are weighted at 0.2 (5 nonresident facilities equals one small resident facility).

3. Higher Level Averages. Scores are also averaged at the branch, division and regional levels by accumulating individual facility scores at each level using the same weighting factors used to compute sectional averages. In the report, the branch averages appear at the end of the last section of the branch. Likewise, the division averages appear at the end of the last branch. The regional averages are found on the last page of the report for the month of record.

III. VALIDATION. The SAG decided that it was necessary to verify that the QUEST model produces reasonably reliable indicators of effectiveness. Unfortunately, there are no measures of effectiveness that are currently being used to compare against the QUEST results. Therefore, validation was attempted by comparing the model with expert opinion. Questionnaires were sent to QA Program Experts at all echelons from Headquarters DLA down to section chiefs in DCASRs ATL, CHI and STL. Each echelon was asked to evaluate the effectiveness and effectiveness trend of each subordinate activity on scales of 0-100. Questionnaire results were converted to ordinal rankings. These rankings were compared with the corresponding QUEST rankings using a statistical technique called the Spearman Rank Correlation Test [3].

A. Initial comparisons were mostly favorable but several instances of high disagreement between model and expert were observed. Through a process of follow-up talks with experts, questionnaire clarification, model changes and revalidation, the revised results of the tests are shown in Table 5. The major factors in disagreement found were misinterpretation of the questionnaire and the use of data external to QAMIS by the expert to evaluate effectiveness.

Table 5

AVERAGE CORRELATION OF QUEST TO EXPERTS

Spearman Rank Correlation-R

Expert	Sub Org Evaluated	Avg # of Subs	Avg Relative Eff R	Prob of Coincidental Correlation	Avg Trend R	Prob of Coincidental Correlation
DLA-Q	DCASR	5	.500	16%	.600	12%
DCASR's	Divisions	9	.366	13%	.265	22%
Divisions	Branches	5	.485	16%	.443	19%
Branches	Sections	5	.267	30%	.300	27%
Section	Facilities	15	.441	5%	.265	16%
Average		8	.412	13%	.375	15%

B. If the model produced the same ranking of subordinates as the expert, the R values shown would be +1.0. If the model exactly inverted the expert's rankings, R would be -1.0. A correlation of zero implies no correlation, neither favorable or contrary. Whether or not a given value of R is significant depends on the number of items ranked. For example, a R value achieved by ranking 5 DCASRs yielding R = .5 could occur by coincidence 16% of the time. The same .5 R value achieved by ranking 9 DCASRs could only occur by coincidence about 7% of the time.

C. The correlation with expert opinion shown in Table 5, although not extremely high, is consistently positive and consistent through the echelons of the organization. In several cases, the experts indicated that they had more confidence in the model than their own assessments.

IV. BENEFITS. The SAG was asked to provide a list of potential benefits of implementing QUEST at various levels of the organization. Results are shown in Table 6. In summary, QUEST provides measures of QA effectiveness that are currently not available. These measures are systematic and informative and they provide QA personnel with insight regarding potential problems and trends for early resolution. QUEST could reduce the burden of first line supervisors and above concerning review of QAMIS data and the burden of Quality Data Evaluation for some QARs.

Table 6

POTENTIAL BENEFITS OF QUEST

For Section Chiefs

1. Provides a quantifiable measure of effectiveness of the QARs application of CQAP.

2. Provides a quantifiable measure of contractor effectiveness.

3. Concise statistical summary of QAMIS data. Should result in less time spent reviewing QAMIS printouts.

4. Helps identify problem areas for staff assistance and training needs.

5. Resource allocation tool. Could be used to justify requests for additional resources.

6. Tool for new supervisors to get oriented.

7. Facilitates comparisons within section and allows visibility on how section compares with peers DLA-wide.

8. Trend becomes more visible.

For Branch Chiefs

1. All of the above.

2. Provides a means to validate FLS reviews.

3. Provides information on individual FSCMs in a concise form.

4. Provides an objective means to compare section performance.

For Division Chiefs

1. All of the above. *

2. Supports decision on where to provide staff support and audits.

3. Vehicle of communication with other functional areas. QAMIS data is in a form that non-QA personnel can relate to.

4. Assists and validates Pre-Award Surveys. Could result in more desk audits.

For Regional Staff

1. All of the above. *

2. Determine where policy guidance and clarification needed.

3. Identifies areas needed for new training programs.

4. Identifies commodity related problems for corrective action.

For DLA-O Staff

1. All of the above. *

2. Identifies areas for Management Reviews.

* Use of QUEST by high level managers for some of the beneficial applications of lower level managers could be a negative benefit of QUEST. Section and Branch Chiefs (as well as brainstorming team members during summer 1986 meetings) have repeatedly warned that they are apprehensive that QUEST will be used to micromanage CQAP. QUEST could be a burden on the lower level managers if they have to respond to frequent inquiries about specific effectiveness problems on a monthly basis.

V. CONCLUSIONS

A. The QUEST model measures the effectiveness of QA as defined in this study. The model produces both detailed and summary level measures from indicators available in the QAMIS.

B. The QUEST model is valid. It was developed based extensively on input from experts and has been successfully tested.

C. The QUEST model has significant potential benefits. It provides the manager with new and useful information. QUEST could improve the productivity of QA Supervisors and could ease the Quality Data Evaluation burden on some QARs. However, if QUEST is used by managers at high levels to micro-manage or to burden lower level managers with inquiries, it may do more harm than good.

D. The QUEST model will require periodic updates. Significant policy changes, changes to QAMIS, new regions using Phase II and general trends could cause certain portions of QUEST to be out-of-date.

VI. RECOMMENDATIONS AND IMPLEMENTATION

A. It is recommended that DLA-Q approve the SAG plan to implement QUEST on a trial basis at one or more Phase II regions for a four to six month test. This test will require assistance from the DLA Systems Automation Center, DSAC. If the QUEST model is determined to be cost effective, by the SAG, then further implementation should follow:

1. All Phase II DCASRs should implement QUEST immediately.

2. DCASRs BOS, LA, NY and PHI should implement QUEST six months following Phase II conversion.

3. A Systems Change Request should be initiated to DSAC to incorporate (reprogram) QUEST into QAMIS. Outputs from QUEST should become a part of the QAMIS performance history file in the future.

4. The QAMIS element called Performance Indicator (PI) should be renamed Productivity Indicator to avoid confusion with QUEST.

5. If QUEST is implemented, managers must avoid a natural tendency to abuse the tool.

B. It is recommended that DLA-LO:

1. Continue to support the QUEST model until a transition to DSAC can be made.

2. Assist DLA-Q in testing QUEST in future installations.

C. The new QAMIS offers opportunities for further analysis of CQAP. It is recommended that further research be sponsored in the following areas:

1. Develop a model, which uses the QUEST process, such as stratification, problem facility identification, etc., to analyze efficiency measures.

2. Establish relationships between resources, efficiency and effectiveness. Using historical data, determine the marginal effectiveness associated with varying workloads and resources.

3. Develop CQAP guidelines which optimize effectiveness. For example, for a given grouping of facilities, the percentage of time allocated to CQAP elements that achieves the highest QUEST scores could be used by QARs to decide the appropriate balance between Planning, PE and PVI.

Appendix A

Study Advisory Group Members

Full Time Members

Name	Organization
Mr. Ronald DiPadova	DLA-QR
Mr. Richard Zerilli	DLA-QR
Mr. Jerry Andrews	DCASR-ATL-Q
Mr. Silvio Pontarelli	DCASR-CHI-Q
Mr. Keith Morrison	DCASR-STL-Q
Mr. Ted Ingalls	DLA-LO

Other Participants

COL Wayne Heard, USA	DLA-QR
CAPT M. Kubishen, USN	DLA-QO
MAJ Brad Wooten, USA	DLA-QO
Mr. Robert Schmitt	DLA-QR
Ms. Dawn Lyon	DLA-QR
CDR John Thompson, USN	DCASR-ATL-Q
Mr. Joseph Jones	DCASR-ATL-Q
Mr. Al Gunsel	DCASR-CHI-Q

Appendix B

Record Layouts

DATA ELEMENT FIELD NAMES

PERFORMANCE HISTORY FILE

DASC-D DATA NAMES	COLUMBUS INPUT SCREEN NAMES
DCASR	DCASR-CD
ORGANIZATION	QA-ORG-CD
FSCM	FSCM
TYPE FACILITY	TY-FAC-CD
PRIMARY COMMODITY	QA-CMDTY-CD-P
PRIMARY PROVISION	QA-PVN-CD-P
REPORTING DATE	RPT-MTH-YR
PLANNING HOURS	PLNG-HRS
LOTS INSPECTED	LOTS-INSP-CNT
LOTS REJECTED	LOTS-REJ-CNT
UNITS INSPECTED	UNITS-INSP
UNITS DEFECTIVE	UNITS-FND-DFCT
INSPECTION HOURS	PROD-INSP-HRS
PE LOCATIONS EVALUATED	PROC-EVAL-LOC-EVAL
PE HOURS	PROC-EVAL-HRS
PROCEDURES REVIEWED	PROC-RVU
PROCEDURE REVIEW HOURS	PROC-RVU-HRS
QDR TYPE A	QDR-TY-A
QDR TYPE B	QDR-TY-B
QDR TYPE C	QDR-TY-C
QDR TYPE D	QDR-TY-D
QDR TYPE E	QDR-TY-E
TOTAL QDR HOURS	TOT-QDR-HRS
TRAVEL HOURS	TRVL-HRS
TRAINING HOURS	TRNG-HRS
SUPPLEMENTAL HOURS	SUPPL-HRS-AVAIL
FMS DOLLARS SHIPPED	FMS-DOL-SHPD
FMS HOURS	FMS-HRS
ADMINISTRATIVE HOURS	ADMIN-SPT-HRS
NUMBER OF SHIPMENTS	SHI-CNT
NUMBER OF WAIVER DEVIATION	WAVR-DEVN-CNT
WAIVER DEVIATION HOURS	WAVR-DEVN-HRS
INTENSIFIED INSPECT HOURS	INTSFD-PROD-INSP-HRS
REINSPECTION HOURS	RINSPCTN-HRS
NONRESIDENT VISITS	NON-RSDNT-VST-CNT

MRB HOURS	MRB-HRS
ECP REVIEWED	ECP-RVU
ECP REVIEW HOURS	ECP-HRS
MDR HOURS	MDR-ACTN-HRS
PCO VISITS	PCO-VST
CONTRACTOR MEETING	CONTRR-MTG-HRS
OUT PO REVIEWED	OUTGNG-PO-RVUD
TOTAL MRB	TOT-MRB
OUT PO HOURS	OUTGNG-PO-RVU-HRS
PCO REQUESTS	PCO-ACTN-RQST
CAO REQUESTS	CAO-ACTN-RQST
AVAILABLE MANHOURS	AVAIL-MANHR
PERFORMANCE INDICATOR	PFMC-IND
SYSTEM INDICATOR	SYS-IND
ESTIMATED PROCESS AVERAGE	EST-PROC-AVG
LOTS REJECTED RATIO	LOTS-REJ-RATIO
VOLUME FACTOR	VOL-FCTR
CONTRACTS RECEIVED	CONTR-CNT-RCVD
CONTRACTS COMPLETED	CONTR-CNT-CMPL
DOLLAR VAL RECEIVED	DV-RCVD
CUR MONTH DOLLARS SHIPPED	DV-SHPD-CURR-MTH
CONTRACTS TYPE A RECEIVED	CONTR-CNT-QA-PVN-A
CONTRACTS TYPE B RECEIVED	CONTR-CNT-QA-PVN-B
CONTRACTS OTHER RECEIVED	CONTR-CNT-OTH
QALI RECEIVED DATE	CONTR-CNT-QA-RMK42
QALI RESCINDED DATE	CONTR-CNT-QA-RMK46
MDR RECEIVED	MDR-RCVD
CONTRACTS TYPE A ONHAND	CONTR-CNT-OH-CNT-PVN-A
CONTRACTS TYPE B ONHAND	CONTR-CNT-OH-CNT-PVN-B
CONTRACTS OTHER ONHAND	CONTR-CNT-OH-OTH
UNDELIVERED DOLLARS	UNDEL-DOL-BAL
NON QAR QDRS	QDR-NQAR
HOURS OVER REQUEST	HRS-OVR-RQST
NUMBER OVER REQUEST	NO-OVR-RQST
DOLLARS OVER REQUEST	DOL-OVR-RQST
UNITS NOTINSPECTED	NO-PVI-SMPL
LOCATIONS NOTEVALUATED	PE-LOC-NOTEVAL
PE ELEMENTS NOTEVALUATED	PE-ELMTS-NOTEVAL
PE ELEMENTS EVALUATED	PE-ELMTS-EVAL

NEW LAYOUT - HAS LARGER FIELD SIZES

RECORD LAYOUT

PROGRAM TITLE	PREPARED BY	DATE
DLA-QHQ; QA DATABASE	DLA-GRS	10CT 85

PROGRAM NUMBER	FILE NAME	TYPE LABEL
	"PH" PERFORMANCE HISTORY	

RECORD TYPE (Check One)	LABEL	RECORD SIZE	BLOCKING FACTOR
☐ CARD ☐ TAPE ☑ DISK			

FIELD NAME/DESCRIPTION	NUMBER OF BYTES	FIELD LOCATION FROM	FIELD LOCATION TO	PICTURE	USAGE
* DCASR-CD	6AN	1	6	6AN	
* QA-CRC-CD	3 AN	7	9	3AN	
* FSCM	6AN	10	15	6AN	
* TY-FAC-CD	1A	16	16	1A	
* QA-CMDTY-CD-P	2AN	17	18	2AN	
* QA-PVN-CD-P	1A	19	19	1A	
* RPT-MTH-YR	4N	20	23	4N	
PLNR--HRS	4N	24	27	4N	
LOTS-INSP-CNT	4N	28	31	4N	
LOTS-REJ-CNT	4N	32	35	4N	
UNITS-INSP	6N	36	41	6N	
UNITS-FND-DFCT	4N	42	45	4N	
PRCD-INSP-HRS	4N	46	49	4N	
PROC-EVAL-LOC-EVAL	3N	50	52	3N	
PROC-EVAL-HRS	4N	53	56	4N	
PROC-RVU	3N	57	59	3N	
PROC-RVU-HRS	4N	60	63	4N	
QDR-TY-A	3N	64	66	3N	
QDR-TY-B	2N	67	68	2N	
QDR-TY-C	2N	69	70	2N	
QDR-TY-D	1N	71	71	1N	
QDR-TY-E	1N	72	72	1N	
TOT-QDR-HRS	4N	73	76	4N	

DSC FORM APR 72 177 EDITION OF JUL 71 MAY BE USED UNTIL EXHAUSTED

RECORD LAYOUT

PROGRAM TITLE		PREPARED BY	DATE
PROGRAM NUMBER	FILE NAME		TYPE LABEL
RECORD TYPE (Check One) ☐ CARD ☐ TAPE ☐ DISK	LABEL	RECORD SIZE	BLOCKING FACTOR

FIELD NAME/DESCRIPTION	NUMBER OF BYTES	FIELD LOCATION FROM	FIELD LOCATION TO	PICTURE	USAGE
TRVL-HRS	4 N	77	80	4N	
TRNG-HRS	4 N	81	84	4N	
SUPPL-HRS-AVAIL	4 N	85	88	4N	
FMS-DOL-SHPD	10 N	89	98	10N	
FMS-HRS	4 N	99	102	4N	
ADMIN-SPT-HRS	4 N	103	106	4N	
SHP-CNT	4 N	107	110	4N	
WAVR-DEVN-CNT	2 N	111	112	2N	
WAVR-DEVN-HRS	4 N	113	116	4N	
INTSFD-PROD-INSP-HRS	3 N	117	119	3 N	
RINSPCTN-HRS	3 N	120	122	3 N	
NON-RSDNT-VST-CNT	2 N	123	124	2N	
MRB-HRS	4 N	125	128	4N	
ECP-RVU	3 N	129	131	3N	
ECP-HRS	4 N	132	135	4N	
MDR-ACTN-HRS	4 N	136	139	4N	
PCO-VST	2N	140	141	2N	
CONTRR-MTG-HRS	4N	142	145	4 N	
OUTGNG-PO-AVUD	3N	146	148	3 N	
TDT-MRB	4 N	149	152	4N	
OUTGNG-PO-RVU-HRS	3N	153	155	3 N	
PCO-ACTN-RQST	2 N	156	157	2N	
CAU-ACTN-RQST	2N	158	159	2N	

DSC FORM 177 APR 72 EDITION OF JUL 71 MAY BE USED UNTIL EXHAUSTED

RECORD LAYOUT

PROGRAM TITLE	PREPARED BY	DATE

PROGRAM NUMBER	FILE NAME	TYPE LABEL

RECORD TYPE (Check One) ☐ CARD ☐ TAPE ☐ DISK	LABEL	RECORD SIZE	BLOCKING FACTOR

FIELD NAME/DESCRIPTION	NUMBER OF BYTES	FIELD LOCATION FROM	FIELD LOCATION TO	PICTURE	USAGE
AVAIL-MAINHR	5 N	160	164	5N	
PRMC-IND	5 N	165	169	999.99	
SYS-IND	5N	170	174	999.99	
EST-PRCC-AVG	5 N	175	179	999.99	
LCTS-REJ-RATIO	5 N	180	184	999.99	
VOL-FCTR	5 N	185	189	999.99	
CONTR-CNT-RCVD	6 N	190	195	6N	
CONTR-CNT-CMPL	6N	196	201	6N	
DV-RCVD	12N	202	213	12N	
DV-SHPD-CURR-MTH	12 N	214	225	12N	
CONTR-CNT-QA-PVN-A	6 N	226	231	6N	
CONTR-CNT-QA-PVN-B	6N	232	237	6N	
CONTR-CNT-OTH	6 N	238	243	6N	
CONTR-CNT-QA-RMK42	6 N	244	249	6N	
CONTR-CNT-QA-RMK46	6N	250	255	6N	
MDR-RCVD	4N	256	259	4N	
CONTR-CNT-CH-CNT-PVN-A	6N	260	265	6N	
CONTR-CNT-CH-CNT-PVN-B	6N	266	271	6N	
CONTR-CNT-CH-OTH	6N	272	277	6N	
UNDEL-DCL-BAL	12N	278	289	12N	
QDR-NQAR	2N	290	291	2N	
HRS-OVR-RQST	3N	292	294	3N	
NC-OVR-RCST	4N	295	298	4N	

DSC FORM 177 APR 72 EDITION OF JUL 71 MAY BE USED UNTIL EXHAUSTED

RECORD LAYOUT

PROGRAM TITLE		PREPARED BY		DATE
PROGRAM NUMBER	FILE NAME			TYPE LABEL
RECORD TYPE (Check One) ☐ CARD ☐ TAPE ☐ DISK		LABEL	RECORD SIZE	BLOCKING FACTOR

FIELD NAME/DESCRIPTION	NUMBER OF BYTES	FIELD LOCATION FROM	FIELD LOCATION TO	PICTURE	USAGE
DOL-OVR-RQST	12 N	299	310	12N	
NO-PVT-SMPL	5 N	311	315	5N	
PE-LOC-NOTEVAL	3 N	316	318	3N	
PE-ELMTS-NOTEVAL	5 N	319	323	5N	
PE-ELMTS-EVAL	5 N	324	328	5N	

BSC FORM 177 APR 72. EDITION OF JUL 71 MAY BE USED UNTIL EXHAUSTED

DATA ELEMENT FIELD NAMES

FACILITY PROFILE FILE

DASC-D DATA NAMES	COLUMBUS INPUT SCREEN NAMES
DCASR	DCASR-CD
ORGANIZATION	QA-ORG-CD
FSCM	FSCM
TYPE FACILITY	TY-FAC-CD
PRIMARY COMMODITY	QA-CMDTY-CD
PRIMARY PROVISION	QA-PVN-CD-P
CONTRACTOR QA PERS	NO-CONTRR-QA
CONTRACTOR PROD PERS	CONTRR-PROD
NUMBER PE LOCATIONS	NO-PROC-EVAL-LOC
GOVERNMENT QA PERS	NO-GOVT-QA
OPERATION TYPE	OPER-TY-CD
MILEAGE	MILGE
SPECIAL PROGRAM	SPCL-QA-PGM
TYPE EQUIPMENT	TY-EQUIP
SPECIAL PROCESS	SPCL-PROCS
PE ELEMENT	PROC-EVAL-EL
CONTRACTOR QA INSPECTION	TY-CONTRR-INSP-EFRT
GOVT VERIFICATION	GOVT-VERFN
SKILL AREA	SKL-AREA-CD
PRIME CONTRACTOR CNTL SUBS	PRI-CNTRL-SUBS

QAMIS

INCLUDES 1 NEW FIELD & EXPANDED RECORD PAGE 1 OF 1

RECORD LAYOUT

PROGRAM TITLE: DLA-Q HQ QA DATABASE | PREPARED BY: DLA-GRS R. DIPAOLA | DATE: 10-1-85

PROGRAM NUMBER | FILE NAME: FACILITY PROFILE DATA (NPFS) | TYPE LABEL

RECORD TYPE (Check One): ☐ CARD ☐ TAPE ☒ DISK | LABEL | RECORD SIZE | BLOCKING FACTOR

FIELD NAME/DESCRIPTION	NUMBER OF BYTES	FIELD LOCATION FROM	FIELD LOCATION TO	PICTURE	USAGE
DLA-SR-CD	6	1	6	6AN	
QA-ORG-CD	3	7	9	3AN	
FSCM	6	10	15	6AN	
TY-FAC-CD	1	16	16	1A	
QA-CMDTY-CD-P	2	17	18	2AN	
QA-PVN-CD-P	1	19	19	1A	
NO-CONTRR-QA	4	20	23	4N	
CONTRR-PROD	5	24	28	5N	
NO-PROC-EVAL-LOC	3	29	31	3N	
NO-GOVT-QA	2	32	33	2N	
OPER-TY-CD	2	34	35	2A	
MILGE	3	36	38	3N	
SPCL-QA-PGM	6	39	44	6AN	
TY-EQUIP	3	45	47	3A	
SPCL-PROCS	32	48	79	32AN	
PROC-EVAL-EL	80	80	159	80A	
TY-CONTRR-INSP-EFRT	3	160	162	3A	
GOVT-VERFN	4	163	166	4A	
SKL-AREA-CD	5	167	171	5A	
PRI-CNTRL-SUBS	1	172	172	1A	

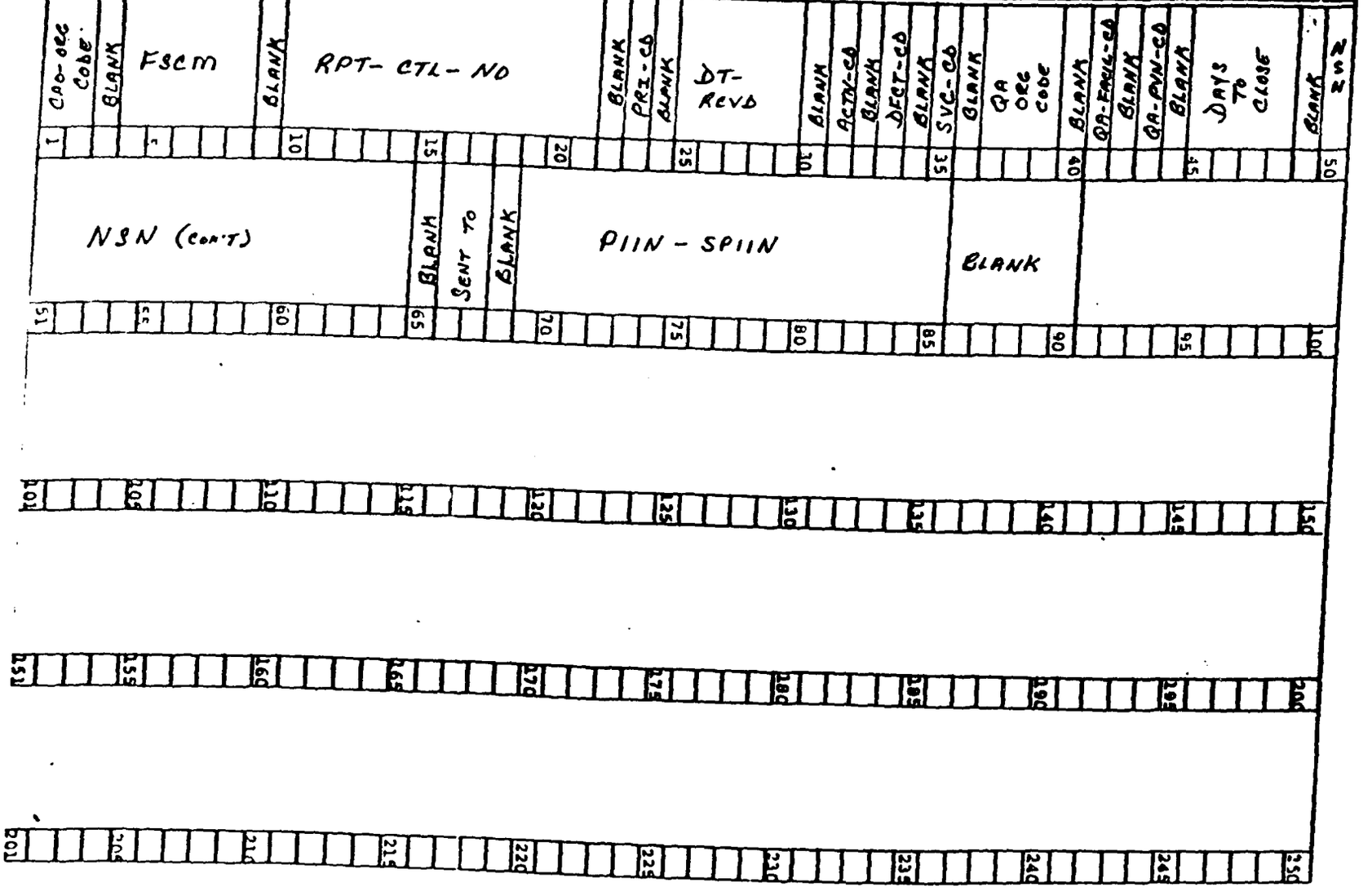
DSC FORM 177 APR 72 EDITION OF JUL 71 MAY BE USED UNTIL EXHAUSTED

PAGE 1 OF 1

FILE/RECORD LAYOUT FORM

DATE 21 OCT 1986

FILE NAME Z9. QAMDRV FILE INDEX NO _____ RECORD SIZE 90 BLOCK SIZE 90 BANNERED Y OR N



Appendix C

MDR Processor

"Valid" MDR's are defined as all MDR's that are completed (Action Code "F"), not for information only (Priority Code = 5) and deficiency is attributable to CQAP (Defect codes = A through M). Reopened and reclosed MDR's are not counted as new valid MDR's, but some of the data from the reopened MDR is moved to the original MDR. For example, if upon reinvestigation, the defect code changes from "U" to "A," the original MDR record will become "valid." The days to close a reopened MDR are added to the days to close the original MDR.

Because there is a time gap associated with CQA actions and MDR receipt, consideration is given to the approximate "age" of the MDR. "Age" represents an estimate of time interval from ship date to MDR receipt date. An MDR for a recent contract carries more weight than a MDR for a five year old contract because the CQA conditions that resulted in the recent contract's MDR probably still exist whereas "old" MDR's probably were caused by conditions that have since been corrected. Age is computed by the MDR processor by subtracting the contract year (from the contract number) from the year the MDR was received. For MDR's with missing or invalid contract numbers, the age is assumed to be 0. Weighting factors were provided by the Study Advisory Group (SAG) using the Social Participatory Allocative Network (SPAN) Technique [1] to elicit expert opinions. A list of Study Advisory Group Members is provided in Appendix C. Results are given in Table C-1.

Table C-1

MDR WEIGHTING FACTORS

Age of MDR (Yr)	Weight
0 - 1	1.00
2	.82
3	.47
4	.18
5 +	.00

The MDR processor creates an MDR record sorted by FSCM by year received by month received in ascending order. The module sums the weighted number of valid MDR's received during the month by the FSCM and computes the average days to close the MDR.

For example, suppose a facility received five MDR's in a particular month as shown in Table C-2. The MDR processor would:

1. exclude the first and third MDR because of "invalid" action code and defect code respectively.

2. replace the contract number in the second MDR with the contract number shown in the reopened MDR and add the days to close the reopened MDR to the original MDR. Delete reopened MDR.

3. convert the Julian Date to month-year format.

4. compute the weighted number of MDR's and compute the average day to close. The record created for the FSCM would show for the month and year (01, 87) a weighted MDR count of 1.00 + .47 = 1.47 and an average day to close of ((30 + 14) + (40)) / 2 = 42 days.

Table C-2

EXAMPLE OF MDR FILE

FSCM	RCN	PRI	DATE	ACT	DEF	DAYS	CONTRACT
ABCDE	SXXXXA870001	4	87002	A	D		DLAXXX84
ABCDE	SXXXXA870027	4	87007	F	A	30	Unknown
ABCDE	SXXXXA870063	4	87013	F	U	72	FXXXXX81
ABCDE	SXXXXA875027*	4	87110	F	A	14	DAAXXX86
ABCDE	SXXXXA870090	4	87027	F	B	40	NXXXXX84

* Reopened 870027

Because of FORTRAN limitations, there are limits on the MDR file that is accessed. The number of MDR's must not exceed:

1. 100,000 total MDR's.

2. 5,000 "valid" MDR's.

3. 100 "valid" MDR per FSCM per month.

JCL TO PROCESS TAPE

```
//ZSHDRV     JOB 2215,'QAMDR TAPE FILE ',CLASS=C,MSGLEVEL=(1,1),      00020002
//           TYPRUN=HOLD,                                             00030000
//           MSGCLASS=X                                               00040000
                                                                      00060000
//*                                                                   00060000
//STEP1 EXEC  PGM=IEBGENER                                            00160002
//SYSPRINT  DD  SYSOUT=*                                              00170000
//SYSUT1     DD DSN=FSA.QAMDR.YP21,DISP=SHR                           00180002
//******************************************* COPY OF QAMDR.YP21 (DATABASE)
//SYSUT2     DD  DSN=ZS.QAMDRV,DISP=(,CATLG),                         00190002
//**************************************** NEW QAMDR FILE
//           UNIT=DISK,SPACE=(TRK,(100,50),RLSE),                     00200002
//           DCB=(LRECL=90,BLKSIZE=90,RECFM=FB)                       00210002
//SYSIN      DD  *                                                    00220000
   GENERATE     MAXFLDS=15
   RECORD       FIELD=(2,124,,1),                                     C00240003
                FIELD=(5,136,,4),                                     C00250003
                FIELD=(12,43,,10),                                    C00260003
                FIELD=(1,272,,23),                                    C00270003
                FIELD=(5,2,,25),                                      C00280003
                FIELD=(1,89,,31),                                     C00290003
                FIELD=(1,79,,33),                                     C00300003
                FIELD=(1,261,,35),                                    C00310003
                FIELD=(3,150,,37),                                    C00320003
                FIELD=(1,273,,41),                                    C00330003
                FIELD=(1,270,,43),                                    C00340002
                FIELD=(4,30,,45),                                     C0340002
                FIELD=(15,162,,50),                                   C00340002
                FIELD=(2,233,,66),                                    C00340002
                FIELD=(17,186,,69)                                    00340002
/*                                                                    00350002
//********* TAPE CREATED BELOW *****************************          00041000
//STEP2      EXEC PGM=UTC123                                          00090000
//SYSUT1     DD  DSN=ZS.QAMDRV,DISP=SHR                               00100001
//**************************************** NEW QAMDR FILE
//SYSUT2     DD  DSN=DCASR.CLE.MDR,DISP=(,KEEP),UNIT=TAPE,            00130001
//           DCB=(RECFM=FB,LRECL=90,BLKSIZE=90),                      00140001
//           LABEL=(,SL,RETPD=5)                                      00141001
//SYSPRINT DD  SYSOUT=*                                               00150000
//SYSIN      DD  DUMMY                                                00160000
 COPY I=SYSUT1,O=SYSUT2                                               00170000
//*                                                                   00180000
```

```
//GOR60603 JOB (GOGO,GOR),'BROOKS',CLASS=1,MSGCLASS=X                    00010007
//*                                                                      00020001
//STEP1 EXEC FORTVCG,FVREGN=250K,GOREGN=200K,                            00030001
//           PARM.GO='LET,NORES,EP=MAIN,SIZE=500000'                     00040004
//FORT.SYSIN DD *                                                        00050004
      CHARACTER*3 FIL1,MO3                                             00060004
      CHARACTER*5 FSCN1                                                00070001
      CHARACTER*1 FIL2,FIL4,FIL5,FIL6,FRCN,FPRI,FAC,FDEF               00080001
      CHARACTER*12 FIL3                                                00090001
      CHARACTER*2 YR,FY                                                00100004
      CHARACTER*11 FIL7                                                00110004
      CHARACTER*4 FDAYS1                                               00120004
      CHARACTER*26 FIL8                                                00130004
      CHARACTER*14 FIL9                                                00140004
1     FORMAT(A3,A5,A1,A1,A12,A1,A1,A2,A3,A1,A1,A1,A1,A11,A4,A26,A2,A14) 00150004
2     READ(1,1,END=3) FIL1,FSCN1,FIL2,FRCN,FIL3,FPRI,                  00160004
     +     FIL4,YR,MO3,FIL5,FAC,FIL6,FDEF,                             00170004
     +     FIL7,FDAYS1,FIL8,FY,FIL9                                    00180004
      IF (FRCN.EQ.'S') THEN                                            00190004
         IF((FY .LT. '00') .OR. (FY .GT. '99')) FY=YR                  00200004
         IF((FPRI.NE.' ').AND.(FPRI.NE.'0').AND.                       00210004
     +   (((FY.GE.'80').AND.(FY.LE.'89')).OR.((FY.GE.'75').AND.        00220004
     +   (FY.LE.'79')))) THEN                                          00230004
            WRITE(2,1) FIL1,FSCN1,FIL2,FRCN,FIL3,FPRI,                 00240004
     +      FIL4,YR,MO3,FIL5,FAC,FIL6,FDEF,                            00250004
     +      FIL7,FDAYS1,FIL8,FY,FIL9                                   00260004
         ENDIF                                                         00270004
      ENDIF                                                            00280004
      GOTO 2                                                           00290004
3     END                                                              00300004
//*                                                                     00310004
//*                                                                     00320004
//GO.FT01F001 DD DSN=GOR.GROVER.MORSTL.TEMP,                            00330007
//            DISP=SHR                                                  00340004
//GO.FT02F001 DD DSN=GOR.BROOKS.STL.MORTEMP1,                           00350007
//            DISP=(,PASS),                                             00360007
//            DCB=(RECFM=FB,LRECL=90,BLKSIZE=18000),                    00370004
//            UNIT=3380,SPACE=(CYL,(1,1),RLSE),                         00380004
//            VOL=SER=DOR001                                            00390004
//GO.FT06F001 DD SYSOUT=*                                               00400004
//SYSOUT DD SYSOUT=*                                                    00410004
//SYSUDUMP DD SYSOUT=*                                                  00420004
//SYSPRINT DD SYSOUT=*                                                  00430004
//*                                                                     00440004
//STEP2   EXEC PGM=ERRC000                                              00450004
//SORTLIB  DD  DSN=SYS1.SORTLIB,DISP=SHR                                00460004
//SYSUDUMP DD  SYSOUT=*                                                 00470001
//SORTMSG  DD  SYSOUT=*                                                 00480004
//SYSOUT   DD  SYSOUT=*                                                 00490004
//SORTIN   DD  DSN=GOR.BROOKS.STL.MORTEMP1,                             00500007
//             DISP=OLD                                                 00510005
//SORTOUT  DD  DSN=GOR.BROOKS.STL.MORTEMP2,                             00520007
//             DISP=(,PASS),                                            00530007
//             UNIT=3380,                                               00540004
//             DCB=(RECFM=FB,LRECL=90,BLKSIZE=18000),                   00550004
//             SPACE=(TRK,(15,1),RLSE),                                 00560004
//             VOL=SER=DOR001                                           00570001
//SORTWK01  DD UNIT=WORKD,SPACE=(TRK,10)                                00580001
//SORTWK02  DD UNIT=WORKD,SPACE=(TRK,10)                                00590004
//SORTWK03  DD UNIT=WORKD,SPACE=(TRK,10)                                00600001
//SYSIN DD *                                                            00610004
 SORT FIELDS=(10,8,CH,A,19,3,CH,A,18,1,CH,A)                           00620001
//*                                                                     00630004
```

```
//*
//STEP3 EXEC FORTVCG,FVREGN=2500K,GOREGN=2000K,
//         PARM.GO='LET,NORES,EP=MAIN,SIZE=500000'
//*
//FORT.SYSIN DD *
      CHARACTER*10 FIL1(2)                                              00640004
      CHARACTER*7 CNBRA(2)                                              00650004
      CHARACTER*1 CNBRB(2)                                              00660004
      CHARACTER*3 CNBRC(2)                                              00670004
      CHARACTER*1 FIL2(2)                                               00680004
      CHARACTER*1 PRI(2)                                                00690004
      CHARACTER*1 FIL3(2)                                               00700004
      CHARACTER*2 YR(2)                                                 00710004
      CHARACTER*3 MO3(2)                                                00720004
      CHARACTER*1 FIL4(2)                                               00730004
      CHARACTER*1 FAC(2)                                                00740004
      CHARACTER*1 FIL5(2)                                               00750004
      CHARACTER*1 FDEF(2)                                               00760004
      CHARACTER*11 FIL6(2)                                              00770004
      INTEGER FDAYS(2)                                                  00780004
      CHARACTER*16 FIL7A(2)                                             00790004
      CHARACTER*10 FIL7B(2)                                             00800004
      CHARACTER*2 FY(2)                                                 00810004
      CHARACTER*14 FIL8(2)                                              00820004
    1 FORMAT(A10,A7,A1,A3,A1,A1,A1,A2,A3,A1,A1,A1,A1,                   00830004
     +A11,I4,A16,A10,A2,A14)                                            00840004
      READ(1,1,END=5) FIL1(1),CNBRA(1),CNBRB(1),CNBRC(1),FIL2(1),       00850004
     +PRI(1),FIL3(1),YR(1),MO3(1),FIL4(1),FAC(1),FIL5(1),FDEF(1),       00860004
     +FIL6(1),FDAYS(1),FIL7A(1),FIL7B(1),FY(1),FIL8(1)                  00870004
    2 READ(1,1,END=3) FIL1(2),CNBRA(2),CNBRB(2),CNBRC(2),FIL2(2),       00880004
     +PRI(2),FIL3(2),YR(2),MO3(2),FIL4(2),FAC(2),FIL5(2),FDEF(2),       00890004
     +FIL6(2),FDAYS(2),FIL7A(2),FIL7B(2),FY(2),FIL8(2)                  00900004
      IF((CNBRA(1).EQ.CNBRA(2)).AND.(CNBRC(1).EQ.CNBRC(2))) THEN        00910004
        FDAYS(1)=FDAYS(1)+FDAYS(2)                                      00920004
        CNBRB(1)=CNBRB(2)                                               00930004
        PRI(1)=PRI(2)                                                   00940004
        FAC(1)=FAC(2)                                                   00950004
        FDEF(1)=FDEF(2)                                                 00960004
        FY(1)=FY(2)                                                     00970004
        IF(FIL1(1).LE.' ') FIL1(1)=FIL1(2)                              00980004
        IF(FIL2(1).LE.' ') FIL2(1)=FIL2(2)                              00990004
        IF(FIL3(1).LE.' ') FIL3(1)=FIL3(2)                              01000004
        IF(FIL4(1).LE.' ') FIL4(1)=FIL4(2)                              01010004
        IF(FIL5(1).LE.' ') FIL5(1)=FIL5(2)                              01020004
        IF(FIL6(1).LE.' ') FIL6(1)=FIL6(2)                              01030004
        IF(FIL7A(1).LE.' ') FIL7A(1)=FIL7A(2)                           01040004
        IF(FIL7B(1).LE.' ') FIL7B(1)=FIL7B(2)                           01050004
        IF(FIL8(1).LE.' ') FIL8(1)=FIL8(2)                              01060004
        GOTO 2                                                          01070004
      ENDIF                                                             01080004
      WRITE(2,1) FIL1(1),CNBRA(1),CNBRB(1),CNBRC(1),FIL2(1),            01090004
     +PRI(1),FIL3(1),YR(1),MO3(1),FIL4(1),FAC(1),FIL5(1),               01100004
     +FDEF(1),FIL6(1),FDAYS(1),FIL7A(1),FIL7B(1),FY(1),FIL8(1)          01110004
      FIL1(1)=FIL1(2)                                                   01120004
      CNBRA(1)=CNBRA(2)                                                 01130004
      CNBRB(1)=CNBRB(2)                                                 01140004
      CNBRC(1)=CNBRC(2)                                                 01150004
      FIL2(1)=FIL2(2)                                                   01160004
      PRI(1)=PRI(2)                                                     01170004
      FIL3(1)=FIL3(2)                                                   01180004
      YR(1)=YR(2)                                                       01190004
      MO3(1)=MO3(2)                                                     01200004
      FIL4(1)=FIL4(2)                                                   01210004
      FAC(1)=FAC(2)                                                     01220004
      FIL5(1)=FIL5(2)                                                   01230004
      FDEF(1)=FDEF(2)                                                   01240004
      FIL6(1)=FIL6(2)                                                   01250004
```

```
      FDAYS(1)=FDAYS(2)
      FIL7A(1)=FIL7A(2)
      FIL7B(1)=FIL7B(2)
      FY(1)=FY(2)
      FIL8(1)=FIL8(2)
      GOTO 2
3     WRITE(2,1) FIL1(1),CNBRA(1),CNBRB(1),CNBRC(1),FIL2(1),
     *PRI(1),FIL3(1),YR(1),MO3(1),FIL4(1),FAC(1),FIL5(1),
     *FDEF(1),FIL6(1),FDAYS(1),FIL7A(1),FIL7B(1),FY(1),FIL8(1)
5     END
/*
//*
//GO.FT01F001 DD DSN=GOR.BROOKS.STL.MDRTEMP2,
//             DISP=OLD
//GO.FT02F001 DD DSN=GOR.BROOKS.STL.MDRTEMP3,
//             DISP=(,PASS),
//             DCB=(RECFM=FB,LRECL=90,BLKSIZE=18000),
//             UNIT=3380,SPACE=(CYL,(1,1),RLSE),
//             VOL=SER=DOR001
//GO.FT06F001 DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
/*
//*
//STEP4 EXEC FORTVCG,FVREGN=2500K,GOREGN=2000K,
//             PARM.GO='LET,NORES,EP=MAIN,SIZE=500000'
//
//FORT.SYSIN DD *
C ***********************************************************************
C
C     MODIFIED BY TLB ON 10/20/87
C
C     INPUT:  MATERIAL DEFICIENCY REPORT (MDR) FILES FROM FIVE DCASRS
C     OUTPUT: LISTINGS OF NUMBER OF MDRS BY MANUFACTURERS BY MONTH
C
C ***********************************************************************
C     DOCUMENTATION SECTION
C ***********************************************************************
C
C      THE PURPOSE OF THIS PROGRAM IS TO CREATE AN INPUT FILE FOR
C          ADDITIONAL PROCESSING.  THE END RESULT OF THE
C          PROGRAM IS A LISTING WHICH SHOWS HOW MANY MATERIAL
C          DEFICIENCY REPORTS (MDR) ARE RECEIVED BY THE DCASR
C          FOR A GIVEN MANUFACTURER IN A GIVEN MONTH.  THE LENGTH
C          OF TIME BETWEEN THE FISCAL YEAR OF THE CONTRACT AND THE
C          DATE THE MDR WAS RECEIVED IS TAKEN INTO CONSIDERATION,
C          WITH THE SHORTER TIME CARRYING THE GREATER WEIGHT.
C          IN ADDITION, THE NUMBER OF DAYS TAKEN TO CLOSE THE MDR
C          ARE LISTED.
C
C     THE LOGIC OF THE MAIN PROGRAM IS AS FOLLOWS:
C
C     FIRST, IN THE SUBROUTINE CALLED "REVIEW", THE RAW DATA IS READ
C          FROM FILE ONE, SCREENED FOR ERRONEOUS DATA FIELDS
C          AND THEN REWRITTEN TO FILE TWO.
C
C     SECOND, IN "READER", FILE TWO IS READ INTO AN ARRAY.
C
C     THIRD, IN "JDATER", THE LAST THREE DIGITS OF THE JULIAN DATE
C          OF THE DATE MDR WAS RECEIVED IS CONVERTED INTO MONTHS
C          ONE THROUGH TWELVE.
C
C     FOURTH, IN "SORTER", THE MDRS ARE SORTED ACCORDING TO FSCM
C           AND THE YEAR AND MONTH THE MDRS WERE RECEIVED.
C
C     FIFTH, IN "WEIGHT", THE TIME BETWEEN THE FISCAL YEAR OF THE
```

```
C          CONTRACT AND THE DATE THE MDR WAS RECEIVED IS                  01960004
C          COMPUTED TO REPRESENT THE AGE OF THE MDR.  THE AGE             01970004
C          IS USED TO WEIGHT THE MDR (SEE VARIABLE DICTIONARY FOR         01980004
C          WEIGHTS USED)                                                  01990004
C                                                                         02000004
C     SIXTH, IN "WRITER", THE FSCM, MONTH AND YEAR THE MDR WAS RECEIVED.  02010004
C          THE WEIGHTED AVERAGE OF NUMBERS OF MDRS AND THE NUMBER OF      02020004
C          DAYS REQUIRED TO CLOSE THE MDR ARE WRITTEN TO FILE THREE.      02030004
C                                                                         02040004
C*****************************************************************      02050004
C     VARIABLE DICTIONARY                                                 02060004
C*****************************************************************      02070004
C                                                                         02080004
C     THE PROGRAM VARIABLES ARE DEFINED AS FOLLOWS:                       02090004
C                                                                         02100004
C     NAME     MEANING                    REMARKS                       02110004
C     ----     -------                    -------                       02120004
C     AC       ACTION CODE                ONE LETTER CODE               02130004
C     DAYSCL   DAYS TO CLOSE              FOUR INTEGER NUMBER OF DAYS   02140007
C                                           TAKEN TO CLOSE OUT MDR      02150004
C     DEF      DEFECT CODE                ONE LETTER CODE               02160004
C     FAC      ACTION CODE                                              02170004
C     FDAYS    DAYS TO CLOSE                                            02180007
C     FDAYS1   DAYS TO CLOSE                                            02190007
C     FDEF     DEFECT CODE                                              02200004
C     FIL1     FILLER #1                  FILLERS WERE USED TO SIMPLIFY 02210004
C     FIL2     FILLER #2                    RECORD MANIPULATIONS        02220004
C     FIL3     FILLER #3                                                02230004
C     FIL4     FILLER #4                                                02240004
C     FIL5     FILLER #5                                                02250004
C     FIL6     FILLER #6                                                02260004
C     FIL7     FILLER #7                                                02270004
C     FIL8     FILLER #8                                                02280004
C     FIL9     FILLER #9                                                02290004
C     FPRI     PRIORITY CODE                                            02300004
C     FRCN     RECORD CONTROL NUMBER      FIRST LETTER IS 'S' IN ALL CASES 02310004
C     FSCM     MANUFACTURER CODE          IDENTIFIES MANUFACTURER       02320004
C     FSCM1    MANUFACTURER CODE                                        02330004
C     FY       FISCAL YEAR                YEAR OF CONTRACT              02340004
C     IEND     END SWITCH                 SET TO '1' IN READER          02350004
C                                           TO PROPERLY END DATA READ IN 02360004
C     IFY      FISCAL YEAR                                              02370004
C     ITEMP1   TEMPORARY STORAGE          USED IN "SORTER"              02380004
C     ITEMP2   TEMPORARY STORAGE          USED IN "SORTER"              02390004
C     ITEMP3   TEMPORARY STORAGE          USED IN "SORTER"              02400004
C     ITEMP4   TEMPORARY STORAGE          USED IN "SORTER"              02410004
C     ITEMP5   TEMPORARY STORAGE          USED IN "SORTER"              02420004
C     ITEMP6   TEMPORARY STORAGE          USED IN "SORTER"              02430004
C     JJ       ARRAY PARAMETER            USED IN "WEIGHT" TO SET ARRAY 02440004
C                                           PARAMETERS IN AGER          02450004
C     JN       DO LOOP PARAMETER          USED IN "SORTER"              02460004
C     KYR      YEAR MDR RECEIVED                                        02470004
C     MDRCTR   MDR COUNTER                THE NUMBER OF VALID MDR RECORDS 02480004
C     MDRNUM   NUMBER OF MDRS             THE NUMBER OF MDR RECORDS INPUT 02490004
C     MDRVAL   NUMBER OF MDRS                                          02500004
C     MO1      MONTH MDR RECEIVED         THREE DIGIT INTEGER           02510004
C     MO2      MONTH MDR RECEIVED         TWO DIGIT CONVERTED INTEGER   02520004
C     MO3      MONTH MDR RECEIVED         THREE CHARACTER VARIABLE USED 02530004
C                                           IN "REVIEW" TO SCREEN DATA   02540004
C     NN       DO LOOP PARAMETER            USED IN "SORTER"            02550004
C     PRI      PRIORITY CODE                                            02560004
C     RCN      RECORD CONTROL NUMBER                                    02570004
C     SORKEY   SORT KEY                                                 02580004
C     WT       WEIGHT                     ARBITRARY WEIGHT ASSIGNED IN  02590004
C                                           "WEIGHT" FOR AGE OF MDRS    02600004
C                                                                       02610004
C     WTDMDR   WEIGHTED NUMBER OF MDRS
```

```
C     YR          YEAR MOR RECEIVED          TWO CHARACTER VARIABLE USED IN    02620004
C                                              "REVIEW"                        02630004
C                                                                              02640004
C                                                                              02650004
C.....                                                                         02660004
C     INITIALIZATION SECTION                                                   02670004
C.....                                                                         02680004
C                                                                              02690004
C     PURPOSE: TO INITIALIZE, DECLARE AND DEFINE PROGRAM VARIABLES             02700004
C                                                                              02710004
      CHARACTER SORKEY(5000)*10,FSCM(5000)*5                                   02720004
      CHARACTER*1 RCN(5000),PRI(5000),DEF(5000),AC(5000)                       02730004
      INTEGER KYR(5000),MO1(5000),MO2(5000),FY(5000),DAYSCL(5000)              02740004
      REAL WTOMOR(5000)                                                        02750004
      COMMON DAYSCL,FY,KYR,MO1,MO2,WTOMOR                                      02760005
      COMMON FSCM,RCN,PRI,AC,DEF,SORKEY                                        02770004
      MORNUM = 10000                                                           02780004
C                                                                              02790004
C.....                                                                         02800004
C     MAIN PROGRAM                                                             02810004
C.....                                                                         02820004
C                                                                              02830004
      CALL REVIEW(MORNUM,MORVAL)                                               02840004
      CALL READER(MORVAL)                                                      02850004
      CALL UDATER(MORVAL)                                                      02860004
      CALL SORTER(MORVAL)                                                      02870004
      CALL WEIGHT(MORVAL)                                                      02880004
      CALL WRITER(MORVAL)                                                      02890004
      STOP                                                                     02900004
      END                                                                      02910004
C                                                                              02920004
C.....                                                                         02930004
C     SUBROUTINES                                                              02940004
C.....                                                                         02950004
      SUBROUTINE REVIEW(MORNUM,MORVAL)                                         02960004
C.....                                                                         02970004
C                                                                              02980004
C     PURPOSE:  TO REVIEW RECORD FOR ERRONEOUS DATA INPUT                      02990004
C               IF THE DATA MEETS THESE CRITERIA THE RECORD WILL               03000004
C               WRITTEN TO FILE NUMBER 2                                       03010004
C                                                                              03020004
C                                                                              03030004
      CHARACTER*5 FSCM1                                                        03040004
      CHARACTER*1 FRCN,FPRI,FAC,FDEF                                           03050004
      CHARACTER*2 YR,FY                                                        03060004
      CHARACTER*3 MO3                                                          03070004
      CHARACTER*4 FDAYS1                                                       03080004
C                                                                              03090004
      MORVAL=0                                                                 03100004
      IEND=0                                                                   03110004
      DO 20 I=1,MORNUM                                                         03120004
      IF (IEND.EQ.1) GOTO 25                                                   03130004
      CALL REEDER(FSCM1,MO3,FY,YR,FDAYS1,FRCN,FPRI,FAC,FDEF,IEND)              03140005
C     **** THAT PRIORITY CODE DOES NOT = 5, BLANK OR ZERO                      03150005
      IF((FPRI.NE.'5').AND.(FPRI.NE.' ').AND.(FPRI.NE.'0').AND.                03160004
C     **** THAT ACTION CODE IS F OR Z                                          03170004
     *   ((FAC.EQ.'F').OR.(FAC.EQ.'Z')).AND.                                   03180004
C     **** THAT DEFECT CODE IS A THRU M                                        03190004
     *   (FDEF.LT.'N')) THEN                                                   03200004
      MORVAL=MORVAL+1                                                          03210004
      CALL WRYTER(FSCM1,MO3,FY,YR,FDAYS1,FRCN,FPRI,FAC,FDEF)                   03220004
      ELSE                                                                     03230004
       GO TO 20                                                                03240004
      ENDIF                                                                    03250004
20    CONTINUE                                                                 03260004
25    RETURN                                                                   03270004
      END                                                                      03280004
C                                                                              03290004
C.....                                                                         03300004
```

```
      SUBROUTINE REEDER(FSCM1,MO3,FY,YR,FDAYS1,FRCN,FPRI,FAC,FDEF,IEND) 03330004
C*****************************************************************      03340004
C                                                                        03350004
C     PURPOSE:  INPUTS A RECORD                                          03360004
C                                                                        03370004
      CHARACTER*3 FIL1,MO3                                               03380004
      CHARACTER*5 FSCM1                                                  03390004
      CHARACTER*1 FIL2,FIL4,FIL5,FIL6,FRCN,FPRI,FAC,FDEF                 03400004
      CHARACTER*12 FIL3                                                  03410004
      CHARACTER*2 YR,FY                                                  03420004
      CHARACTER*11 FIL7                                                  03430004
      CHARACTER*4 FDAYS1                                                 03440004
      CHARACTER*26 FIL8                                                  03450004
      CHARACTER*14 FIL9                                                  03460004
C                                                                        03470004
      READ(1,101,END=11) FIL1,FSCM1,FIL2,FRCN,FIL3,FPRI,                 03480004
     +      FIL4,YR,MO3,FIL5,FAC,FIL6,FDEF,                              03490004
     +      FIL7,FDAYS1,FIL8,FY,FIL9                                     03500004
C                                                                        03510004
101   FORMAT(A3,A5,A1,A1,A12,A1,A1,A2,A3,A1,A1,A1,A1,A11,A4,A26,A2,A14)  03520004
      GO TO 12                                                           03530004
11    IEND=1                                                             03540004
12    RETURN                                                             03550004
      END                                                                03560004
C                                                                        03570004
C*****************************************************************      03580004
      SUBROUTINE WRYTER(FSCM1,MO3,FY,YR,FDAYS1,FRCN,FPRI,FAC,FDEF)       03590004
C*****************************************************************      03600004
C                                                                        03610004
C     PURPOSE:  TO WRITE OUTPUT TO FILE NUMBER 2                         03620004
C                                                                        03630004
      CHARACTER*5 FSCM1                                                  03640004
      CHARACTER*2 YR,FY                                                  03650004
      CHARACTER*1 FIL2,FIL4,FIL5,FIL6,FRCN,FPRI,FAC,FDEF                 03660004
      CHARACTER*3 MO3                                                    03670004
      CHARACTER*4 FDAYS1                                                 03680004
      WRITE(2,102) FSCM1,FRCN,FPRI,YR,MO3,FAC,FDEF,FDAYS1,               03690004
     +      FY,FSCM1,YR,MO3                                              03700004
102   FORMAT(A5,A1,A1,A2,A3,A1,A1,A4,A2,A5,A2,A3)                        03710004
      RETURN                                                             03720004
      END                                                                03730004
C                                                                        03740004
C*****************************************************************      03750004
      SUBROUTINE READER(MDRVAL)                                          03760004
C*****************************************************************      03770004
C                                                                        03780004
C     PURPOSE:  INPUTS A RECORD FROM FILE NUMBER 2                       03790004
C                                                                        03800004
      CHARACTER SQRKEY(5000)*10,FSCM(5000)*5                             03810004
      CHARACTER*1 RCN(5000),PRI(5000),DEF(5000),AC(5000)                 03820004
      INTEGER KYR(5000),MO1(5000),MO2(5000),IFY(5000),DAYSCL(5000)       03830004
      REAL WTDMDR(5000)                                                  03840004
      COMMON DAYSCL,IFY,KYR,MO1,MO2,WTDMDR                               03850004
      COMMON FSCM,RCN,PRI,AC,DEF,SQRKEY                                  03860004
      REWIND 2                                                           03870004
C                                                                        03880004
      DO 10 I=1,MDRVAL                                                   03890004
100   READ(2,102) FSCM(I),RCN(I),PRI(I),                                 03900004
     +      KYR(I),MO1(I),AC(I),DEF(I),DAYSCL(I),                        03910004
     +      IFY(I),SQRKEY(I)                                             03920004
102   FORMAT(A5,A1,A1,I2,I3,A1,A1,I4,I2,A10)                             03930004
10    CONTINUE                                                           03940004
20    RETURN                                                             03950004
      END                                                                03960004
C                                                                        03970004
C*****************************************************************      03980004
```

```
      SUBROUTINE JDATER(MORVAL)                                          03990004
C*****************************************************************      04000004
C.                                                                       04010004
C     PURPOSE:  TO CONVERT JULIAN DATES INTO MONTHS 1 - 12               04020004
C                                                                        04030004
C                                                                        04040004
      CHARACTER SORKEY(5000)*10,FSCM(5000)*5                             04050004
      CHARACTER*1 RCN(5000),PRI(5000),DEF(5000),AC(5000)                 04060004
      INTEGER KYR(5000),MO1(5000),MO2(5000),IFY(5000),DAYSCL(5000)       04070004
      REAL WTOMOR(5000)                                                  04080004
      COMMON DAYSCL,IFY,KYR,MO1,MO2,WTOMOR                               04090004
      COMMON FSCM,RCN,PRI,AC,DEF,SORKEY                                  04100004
C                                                                        04110004
      DO 45 J=1,MORVAL                                                   04120004
     1 IF((MO1(J).GE.001).AND.(MO1(J).LE.030)) THEN                      04130004
        MO2(J) = 01                                                      04140004
       ELSE IF((MO1(J).GE.031).AND.(MO1(J).LE.059)) THEN                 04150004
        MO2(J) = 02                                                      04160004
       ELSE IF((MO1(J).GE.060).AND.(MO1(J).LE.089)) THEN                 04170004
       MO2(J) = 03                                                       04180004
       ELSE IF((MO1(J).GE.090).AND.(MO1(J).LE.119)) THEN                 04190004
       MO2(J) = 04                                                       04200004
       ELSE IF((MO1(J).GE.120).AND.(MO1(J).LE.151)) THEN                 04210004
       MO2(J) = 05                                                       04220004
       ELSE IF((MO1(J).GE.152).AND.(MO1(J).LE.180)) THEN                 04230004
       MO2(J) = 06                                                       04240004
       ELSE IF((MO1(J).GE.181).AND.(MO1(J).LE.211)) THEN                 04250004
       MO2(J) = 07                                                       04260004
       ELSE IF((MO1(J).GE.212).AND.(MO1(J).LE.242)) THEN                 04270004
       MO2(J) = 08                                                       04280004
       ELSE IF((MO1(J).GE.243).AND.(MO1(J).LE.272)) THEN                 04290004
       MO2(J) = 09                                                       04300004
       ELSE IF((MO1(J).GE.273).AND.(MO1(J).LE.305)) THEN                 04310004
       MO2(J) = 10                                                       04320004
       ELSE IF((MO1(J).GE.306).AND.(MO1(J).LE.335)) THEN                 04330004
       MO2(J) = 11                                                       04340004
       ELSE IF((MO1(J).GE.336).AND.(MO1(J).LE.366)) THEN                 04350004
       MO2(J) = 12                                                       04360004
      ENDIF                                                              04370004
45    CONTINUE                                                           04380004
50    RETURN                                                             04390004
      END                                                                04400004
C                                                                        04410004
C                                                                        04420004
C*****************************************************************      04430004
      SUBROUTINE SORTER(MORVAL)                                          04440004
C*****************************************************************      04450004
C                                                                        04460004
C     PURPOSE: TO PREPARE FOR LATER MERGING SORTING IS DONE              04470004
C              BY FSCM BY YEAR MOR RECEIVED AND BY MONTH                 04480004
C                                                                        04490004
C                                                                        04500004
      CHARACTER*10 ITEMP6                                                04510004
      CHARACTER*5 ITEMP1                                                 04520004
      CHARACTER SORKEY(5000)*10,FSCM(5000)*5                             04530004
      CHARACTER*1 RCN(5000),PRI(5000),DEF(5000),AC(5000)                 04540004
      INTEGER KYR(5000),MO1(5000),MO2(5000),IFY(5000),DAYSCL(5000)       04550004
      REAL WTOMOR(5000)                                                  04560004
      COMMON DAYSCL,IFY,KYR,MO1,MO2,WTOMOR                               04570004
      COMMON FSCM,RCN,PRI,AC,DEF,SORKEY                                  04580004
      NN=MORVAL-1                                                        04590004
      DO 110 J=1,NN                                                      04600004
      JM = MORVAL-1                                                      04610004
C                                                                        04620004
      DO 120 I=1,JM                                                      04630004
      IF (SORKEY(I).LE.SORKEY(I+1)) THEN                                 04640004
          GO TO 120                                                      04650004
      ELSE                                                               04660004
```

```
                ITEMP1 = FSCM(I)                                        04650004
                ITEMP2 = KYR(I)                                         04660004
                ITEMP3 = MO2(I)                                         04670004
                ITEMP4 = DAYSCL(I)                                      04680004
                ITEMP5 = IFY(I)                                         04690004
                ITEMP6 =SORKEY(I)                                       04700004
                FSCM(I) = FSCM(I+1)                                     04710004
                KYR(I) = KYR(I+1)                                       04720004
                MO2(I) = MO2(I+1)                                       04730004
                DAYSCL(I) = DAYSCL(I+1)                                 04740004
                IFY(I) = IFY(I+1)                                       04750004
                SORKEY(I) = SORKEY(I+1)                                 04760004
                FSCM(I+1) = ITEMP1                                      04770004
                KYR(I+1) = ITEMP2                                       04780004
                MO2(I+1) = ITEMP3                                       04790004
                DAYSCL(I+1) = ITEMP4                                    04800004
                IFY(I+1) = ITEMP5                                       04810004
                SORKEY(I+1) =ITEMP6                                     04820004
             ENDIF                                                      04830004
          120 CONTINUE                                                  04840004
          110 CONTINUE                                                  04850004
              RETURN                                                    04860004
              END                                                       04870004
C                                                                       04880004
C*************************************************************        04890004
              SUBROUTINE WEIGHT(MORVAL)                                 04900004
C*************************************************************        04910004
C                                                                       04920004
C     PURPOSE:  TO COMPUTE A WEIGHTED AVERAGE OF MORS BY MANUFACTURER   04930004
C               BY MONTH MOR WAS RECEIVED                               04940004
C                                                                       04950004
C                                                                       04960004
              INTEGER MORCTR(5000)                                      04970004
              CHARACTER SORKEY(5000)*10,FSCM(5000)*5                    04980004
              CHARACTER*1 RCN(5000),PRI(5000),DEF(5000),AC(5000)        04990004
              INTEGER KYR(5000),MO1(5000),MO2(5000),IFY(5000),DAYSCL(5000) 05000004
              REAL WTOMOR(5000)                                         05010004
              COMMON DAYSCL,IFY,KYR,MO1,MO2,WTOMOR                      05020004
              COMMON FSCM,RCN,PRI,AC,DEF,SORKEY                         05030004
              M=0                                                       05040004
          25  M = M + 1                                                05050004
              IF (M.EQ.MORVAL+1) GO TO 55                               05060004
              WTOMOR(M)=0.0                                             05070004
              CALL SAMER(MORVAL,MORCTR,M)                               05080004
              L=MORCTR(M)                                               05090004
              DO 45 N=1,L                                               05100004
              JJ=M                                                      05110004
              IF (L .EQ. 1) GOTO 30                                     05120004
              JJ = M + N -1                                             05130004
              IF (N .EQ. 1) GOTO 30                                     05140004
              DAYSCL(M) = DAYSCL(M) + DAYSCL(JJ)                        05150004
          30  CALL AGER(WT,JJ)                                         05160004
              WTOMOR(M) = WTOMOR(M)+WT                                  05170004
          45  CONTINUE                                                 05180004
              DAYSCL(M) = DAYSCL(M)/MORCTR(M)                           05190004
              M=M+MORCTR(M)-1                                           05200004
              GO TO 25                                                  05210004
          55  RETURN                                                   05220004
              END                                                       05230004
C                                                                       05240004
C*************************************************************        05250004
              SUBROUTINE SAMER(MORVAL,MORCTR,M)                         05260004
C*************************************************************        05270004
C                                                                       05280004
C     PURPOSE:  TO CHECK FOR RECORDS WITH THE SAME FSCM,                05290004
C               YEAR AND MONTH AND INCREMENT A COUNTER                  05300004
C               CALLED MORCTR TO TRACK THE NUMBER OF RECORDS
```

```
C              WITH THE SAME PARAMETERS.  THIS TOTAL WILL                  05310004
C              BE USED IN WAITER TO COMPUTE WEIGHTED AVERAGES              05320004
C                                                                          05330004
C                                                                          05340004
C                                                                          05350004
      INTEGER MDRCTR(5000)                                                 05360004
      CHARACTER SORKEY(5000)*10,FSCM(5000)*5                               05370004
      CHARACTER*1 RCN(5000),PRI(5000),DEF(5000),AC(5000)                   05380004
      INTEGER KYR(5000),MO1(5000),MO2(5000),IFY(5000),DAYSCL(5000)         05390004
      REAL WTDMDR(5000)                                                    05400004
      COMMON DAYSCL,IFY,KYR,MO1,MO2,WTDMDR                                 05410004
      COMMON FSCM,RCN,PRI,AC,DEF,SORKEY                                    05420004
C                                                                          05430004
C                                                                          05440004
C     THIS DO LOOP IS LIMITED TO THE NUMBER                                05450004
C     OF MDRS OF A GIVEN FSCM IN A MONTH                                   05460004
C     ***NOTE** THE UPPER LIMIT OF THE LOOP MAY REQUIRE MAINTENANCE        05470004
C                                                                          05480004
      MDRCTR(N)=1                                                          05490004
      DO 60 K=1,100                                                        05500004
      IF ((FSCM(N).EQ.FSCM(N+K)).AND.(KYR(N).EQ.KYR(N+K)).AND.             05510004
     +   (MO2(N).EQ.MO2(N+K))) THEN                                        05520004
         MDRCTR(N) = MDRCTR(N) + 1                                         05530004
      ELSE                                                                 05540004
         GO TO 40                                                          05550004
      ENDIF                                                                05560004
60    CONTINUE                                                             05570004
40    RETURN                                                               05580004
      END                                                                  05590004
C                                                                          05600004
C*************************************************************           05610004
      SUBROUTINE AGER(WT,JJ)                                               05620004
C*************************************************************           05630004
C                                                                          05640004
C     PURPOSE:  TO COMPUTE AN AGE AND CORRELATE IT TO A RESPECTIVE         05650004
C               WEIGHT                                                     05660004
C                                                                          05670004
      INTEGER AGE                                                          05680004
      CHARACTER SORKEY(5000)*10,FSCM(5000)*5                               05690004
      CHARACTER*1 RCN(5000),PRI(5000),DEF(5000),AC(5000)                   05700004
      INTEGER KYR(5000),MO1(5000),MO2(5000),IFY(5000),DAYSCL(5000)         05710004
      REAL WTDMDR(5000)                                                    05720004
      COMMON DAYSCL,IFY,KYR,MO1,MO2,WTDMDR                                 05730004
      COMMON FSCM,RCN,PRI,AC,DEF,SORKEY                                    05740004
      WT=0                                                                 05750004
      AGE=KYR(JJ)- IFY(JJ)+ 1                                              05760004
      IF ((AGE.EQ.1).OR.(AGE.EQ.2)) WT = 1.0                               05770004
      IF (AGE.EQ.3) WT = .82                                               05780004
      IF (AGE.EQ.4) WT = .47                                               05790004
      IF (AGE.EQ.5) WT = .18                                               05800004
55    RETURN                                                               05810004
      END                                                                  05820004
C                                                                          05830004
C*************************************************************           05840004
      SUBROUTINE WRITER(MDRVAL)                                            05850004
C*************************************************************           05860004
C                                                                          05870004
C     PURPOSE:  TO WRITE OUTPUT TO FILE NUMBER 3                           05880004
C                                                                          05890004
C                                                                          05900004
      CHARACTER SORKEY(5000)*10,FSCM(5000)*5                               05910004
      CHARACTER*1 RCN(5000),PRI(5000),DEF(5000),AC(5000)                   05920004
      INTEGER KYR(5000),MO1(5000),MO2(5000),IFY(5000),DAYSCL(5000)         05930004
      REAL WTDMDR(5000)                                                    05940004
      COMMON DAYSCL,IFY,KYR,MO1,MO2,WTDMDR                                 05950004
      COMMON FSCM,RCN,PRI,AC,DEF,SORKEY                                    05960004
C                                                                          
C     WRITE THE FIRST RECORD                                               
```

```
      WRITE(3,190) FSCM(I),MO2(I),KYR(I),WTOMOR(I),DAYSCL(I)                05970004
                                                                            05980004
C                                                                           05990004
C     WRITER THE REST OF THE FILE                                           06000004
      DO 192 I=2,MORVAL                                                     06010004
      IF ((FSCM(I).EQ.FSCM(I-1)).AND.(KYR(I).EQ.KYR(I-1))                   06020004
     +    .AND.(MO2(I).EQ.MO2(I-1))) THEN                                   06030004
          GO TO 192                                                         06040004
      ELSE                                                                  06050004
          WRITE(3,190) FSCM(I),MO2(I),KYR(I),WTOMOR(I),DAYSCL(I)            06060004
      ENDIF                                                                 06070004
  190 FORMAT(A5,2X,I2,2X,I3,2X,F5.2,2X,I4)                                  06080004
  192 CONTINUE                                                              06090004
      RETURN                                                                06100004
      END                                                                   06110004
//*                                                                         06120007
//GO.FT01F001 DD DSN=GOR.BROOKS.STL.MORTEMP3,DISP=OLD                       06130007
//GO.FT02F001 DD DSN=GOR.BROOKS.STL.MORTEMP4,                               06140007
//            DISP=(,PASS),                                                 06150004
//            UNIT=WORKD,SPACE=(CYL,(1,10),RLSE),                           06160004
//            DCB=(RECFM=FB,LRECL=30,BLKSIZE=3000),                         06161004
//            VOL=SER=DOR001                                                06170007
//GO.FT03F001 DD DSN=GOR.GROVER.MOR.INSTL.SEP87A,                           06180007
//            DISP=(NEW,CATLG,DELETE),                                      06190004
//            UNIT=3380,SPACE=(CYL,(1,1),RLSE),                             06200004
//            DCB=(RECFM=FB,LRECL=27,BLKSIZE=2700),                         06210004
//            VOL=SER=DOR001                                                06220004
//GO.FT06F001 DD SYSOUT=*                                                   06230004
//SYSOUT DD SYSOUT=*                                                        06240004
//SYSUDUMP DD SYSOUT=*                                                      06250004
//SYSPRINT DD SYSOUT=*                                                      06260004
/*                                                                          06270004
//
```

Appendix D

File Merger

```
//GOR6040F JOB (6040,GOR),'GROVER',CLASS=3,MSGCLASS=X                    00010034
//RUNFTN EXEC FORTVCG                                                    00030005
//FORT.SYSIN DD DSN=GOR.GROVER.FOR(MATCH2),DISP=SHR                      00040024
//GO.FT01F001 DD DSN=GOR.GROVER.STL.SORT.SEP87,DISP=SHR                  00044039
//GO.FT02F001 DD DSN=GOR.GROVER.FAC.STL.SEP87,DISP=SHR                   00045039
//GO.FT03F001 DD DSN=GOR.GROVER.MDR.INSTL.SEP87,DISP=SHR                 00046039
//GO.FT04F001 DD DSN=GOR.GROVER.STL.INTAPE.JUL87,UNIT=TAPE,              00047040
//     DISP=(NEW,CATLG,DELETE),DCB=(LRECL=347,RECFM=FB,BLKSIZE=17350),   00048029
//     LABEL=EXPDT=90001                                                 00049036
//GO.FT06F001 DD SYSOUT=*                                                00049129
//SYSOUT DD SYSOUT=*                                                     00049229
//SYSUDUMP DD SYSOUT=*                                                   00049329
//SYSPRINT DD SYSOUT=*                                                   00049429
//                                                                       00049529
```

```
C   THIS MODULE MERGES THE FACILTY PROFILE, THE HISTORY FILE AND          00010033
C   THE MDR FILE BY FSCM TO PRODUCE A MASTER DATA FILE FOR THE MODEL.     00020033
C                                                                         00030033
C                                                                         00040033
C    DECLARE AND ARRAY VARIABLES                                          00050033
C    ARRAYS ARE DIMENSIONED TO ACCEPT NO MORE THAN TEN YEARS OF DATA       00060033
C                                                                         00070016
      CHARACTER A*5,AA(120)*5,DUM1(120)*10,DUM2(120)*4                    00080010
      CHARACTER DUM3(120)*305,AOPER(120)*2,OPER*2,FLUF1*10                00090013
      CHARACTER FLUF2*16,FLUF3*137,AAA*5                                  00100035
      INTEGER BB(120),CC(120),B,C,E,NOAR(120),DAYSCL(120)                 00110037
      REAL WMDR(120)                                                      00120010
C                                                                         00130010
C     *** INITIALIZE AND SET DEFAULT VALUES ***                           00140033
C                                                                         00150010
      IFSCM=0                                                             00160010
      IYR=0                                                               00170010
      IMONTH=0                                                            00180033
C   NEXT THREE VARIABLES ARE USED TO SUPPRESS FILE READ                   00190012
      ICTL=1                                                              00200021
      JCTL=1                                                              00210021
      KCTL=1                                                              00220031
      AAA='AAAAA'                                                         00230041
      A='AAAAA'                                                           00240020
      ICOUNT=0                                                            00250016
      ISKIP=1                                                             00260016
      DO 5 II=1,120                                                       00270016
       AA(II)='00000'                                                     00280016
       BB(II)=0                                                           00290016
       CC(II)=0                                                           00300016
    5 CONTINUE                                                            00310034
C    SET DEFAULT VALUES                                                   00320010
   10 DO 20 I=1,120                                                       00330012
       NOAR(I)=0                                                          00340035
       WMDR(I)=0.0                                                        00350035
       DAYSCL(I)=0                                                        00360012
       AOPER(I)='AB'                                                      00370010
   20 CONTINUE                                                            00380010
      INPP1=1                                                             00390033
C                                                                         00400010
C     *** INITIALIZATION COMPLETE ***                                     00410010
C                                                                         00420010
C     *** STEP 2 ***                                                      00430010
C     *** READ NPP1 FILE RECORD TO BEGIN MATCHING PROCESS ***             00440033
C                                                                         00450033
C    THE HISTORY FILE IS SKIPPED IF THE FIRST HISTORY RECORD FOR A FSCM   00460033
C    HAS ALREADY BEEN READ.SEE 'RESET COUNTERS AND DEFAULT' SECTION.      00470016
  100 IF(ISKIP-1) 109,101,101                                             00480016
  101 READ(1,102,END=330) DUM1(INPP1),AA(INPP1),DUM2(INPP1),BB(INPP1)     00490010
     1,CC(INPP1),DUM3(INPP1)                                              00500016
  102 FORMAT(A10,A5,A4,I2,I2,A305)                                        00510016
      ISKIP=1                                                             00520033
C    TIMING CONVENTION FOR HISTORY RECORDS IS JNPP1 IS THE CURRENT RECORD 00530033
C    AND KNPP1 IS THE PREVIOUS RECORD. INPP1 IS THE NEXT RECORD TO READ.  00540010
      JNPP1=INPP1                                                         00550010
      KNPP1=INPP1-1                                                       00560010
      INPP1=INPP1+1                                                       00570033
C    UPON IDENTIFICATION OF A FIRST FSCM RECORD, ATTEMPT TO MERGE WITH    00580033
C    FACILITY PROFILE                                                     00590021
      IF(JNPP1 .EQ. 1) GO TO 110                                          00600033
C    IF CURRENT AND PREVIOUS RECORDS HAVE DIFFERENT FSCMS, WRITE PREVIOUS 00610033
C    FSCM RECORDS TO TAPE.IF THE SAME ATTEMPT TO MERGE WITH MDR FILE.     00620022
      IF(AA(JNPP1) .GT. AA(KNPP1)) GOTO 190                               00630012
       IF(AA(JNPP1) .EQ. AA(KNPP1)) THEN
```

```
C       IF(((CC(JNPP1) .GT. C) .OR. (BB(JNPP1) .GT. B)) .AND. (CC(JNPP1)00640039
C     1 .EQ. C)) GOTO 104                                               00650039
        IF((AA(JNPP1) .GT. A) .OR. ((AA(JNPP1) .EQ. A) .AND. ((CC(JNPP1)00660040
     1  .GT. C) .OR. ((BB(JNPP1) .GT. B) .AND. (CC(JNPP1) .EQ. C)))))   00670040
     2  GOTO 104                                                        00680040
        IF((AA(JNPP1) .LT. A) .OR. ((AA(JNPP1) .EQ. A) .AND. ((CC(JNPP1)00690040
     1  .LT. C) .OR. ((BB(JNPP1) .LT. B) .AND. (CC(JNPP1) .EQ. C)))))   00700041
     2  GOTO 101                                                        00710040
         GO TO 240                                                      00720040
        ELSE                                                            00730012
         GO TO 310                                                      00740012
        ENDIF                                                           00750012
  104 ICTL=1                                                            00760021
      GO TO 200                                                         00770021
C                                                                       00780033
C     *** STEP 3 ***                                                    00790033
C     *** READ FACILITY PROFILE FILE ***                                00800033
C                                                                       00810033
  109 INPP1=2                                                           00820016
C  IF END OF PROFILE HAS BEEN REACHED OR HISTORY LAGS PREVIOUS PROFILE  00830033
C  SKIP THIS SECTION                                                    00840033
  110 IF(JCTL .EQ. 0) GOTO 200                                          00850021
      IF(AAA .GT. AA(JNPP1)) GOTO 200                                   00860021
C  IF HISTORY MATCHES PREVIOUS PROFILE, SKIP THE READ AND MERGE RECORDS 00870033
      IF(AAA .EQ. AA(JNPP1)) GOTO 130                                   00880021
C  READ A NEW PROFILE RECORD. IF FSCM MATCHES HISTORY FSCM, MERGE       00890033
C  RECORDS. OTHERWISE REPEAT OR GO LOOK AT MDR FILE                     00900033
      READ(2,120,END=180) FLUF1,AAA,FLUF2,MQAR,OPER,FLUF3               00910021
  120 FORMAT(A10,A5,A16,I2,A2,A137)                                     00920010
C  IF NO MATCH IS FOUND, DEFAULTS VALUES ARE KEPT. SEE INITIALIZATION.  00930033
      IF (AAA .GT. AA(JNPP1)) GOTO 200                                  00940021
       IF(AAA .EQ. AA(JNPP1)) THEN                                      00950012
         GO TO 130                                                      00960012
        ELSE                                                            00970012
         GO TO 110                                                      00980012
        ENDIF                                                           00990012
C  A MATCH HAS BEEN MADE. APPEND PROFILE VARIABLES TO HISTORY RECORD.   01000033
  130 DO 140 J=1,120                                                    01010010
       NQAR(J)=MQAR                                                     01020012
       AOPER(J)=OPER                                                    01030012
  140 CONTINUE                                                          01040010
      GO TO 200                                                         01050028
C  ONCE END OF PROFILE HAS BEEN REACHED, JCTL TURNS OFF FUTURE READS    01060033
  180 JCTL=0                                                            01070030
      GO TO 200                                                         01080023
C                                                                       01090033
C     *** STEP 4 ***                                                    01100033
C     *** WRITE RECORD TO TAPE**                                        01110033
C                                                                       01120033
  190 DO 192 L=1,KNPP1                                                  01130010
       WRITE(4,191) DUM1(L),AA(L),DUM2(L),BB(L),CC(L),DUM3(L),NQAR(L)   01140012
     1 ,AOPER(L),WMDR(L),DAYSCL(L),L,KNPP1                              01150035
  191  FORMAT(A10,A5,A4,I2,I2,A305,I2,A2,F5.2,I4,2I3)                   01160035
       ICOUNT=ICOUNT+1                                                  01170016
       AA(L)='00000'                                                    01180016
       BB(L)=0                                                          01190016
       CC(L)=0                                                          01200016
  192 CONTINUE                                                          01210010
C     *** RESET COUNTERS AND DEFAULTS' ***                              01220010
      AA(1)=AA(JNPP1)                                                   01230016
      BB(1)=BB(JNPP1)                                                   01240016
      CC(1)=CC(JNPP1)                                                   01250016
      DUM1(1)=DUM1(JNPP1)                                               01260016
      DUM2(1)=DUM2(JNPP1)                                               01270016
      DUM3(1)=DUM3(JNPP1)                                               01280016
      JNPP1=1                                                           01290010
```

```
      ISKIP=0                                                          01300016
      GO TO 10                                                         01310010
C     *** END OF STEP 4 ***                                            01320033
C                                                                      01330010
C     *** MATCH TO MDR FILE BY FSCM AND MONTH AND YEAR ***             01340033
C     *** STEP 5 ***                                                   01350033
C                                                                      01360033
C    READ IS SUPPRESSED IF THE END OF MDR FILE HAS BEEN REACHED OR     01370033
C    HISTORY FILE LAGS MDR FILE                                        01380033
  200 IF(ICTL-1) 215,201,201                                           01390012
  201 IF(KCTL .EQ. 0) GOTO 101                                         01400021
C     READ MDR RECORD. PICK UP FSCM, YEAR, MONTH, AND MDR COUNT        01410033
      READ(3,210,ERR=296,END=295) A,B,C,D,E                            01420035
  210 FORMAT(A5,2X,I2,2X,I3,2X,F5.2,2X,I4)                             01430035
  215 IF(A .GT. AA(JNPP1)) GOTO 270                                    01440016
       IF(A .EQ. AA(JNPP1)) THEN                                       01450016
         GO TO 220                                                     01460012
        ELSE                                                           01470012
         ICTL=1                                                        01480038
         GO TO 201                                                     01490016
        ENDIF                                                          01500012
  220 IF(C-CC(JNPP1)) 201,230,280                                      01510035
  230 IF(B-BB(JNPP1)) 201,240,290                                      01520035
C    A MATCH HAS BEEN MADE. APPEND MDR COUNT TO HISTORY RECORD         01530033
  240 WMDR(JNPP1)=D                                                    01540016
      DAYSCL(JNPP1)=E                                                  01550035
      ICTL=1                                                           01560012
      GO TO 101                                                        01570016
C    THE HISTORY FILE LAGS THE MDR FILE. GO BACK AND READ ANOTHER      01580033
C    HISTORY RECORD.MDR COUNT OF CURRENT HISTORY DEFAULTS TO ZERO      01590033
  270 IFSCM=IFSCM+1                                                    01600010
      ICTL=0                                                           01610012
      GO TO 101                                                        01620016
  280 IYR=IYR+1                                                        01630010
      ICTL=0                                                           01640021
      GO TO 101                                                        01650016
  290 IMONTH=IMONTH+1                                                  01660010
      ICTL=0                                                           01670021
      GO TO 101                                                        01680016
  295 KCTL=0                                                           01690023
      GO TO 101                                                        01700021
  296 GO TO 200                                                        01710025
C     *** END OF STEP 5 ***                                            01720033
C                                                                      01730010
C     *** END PROGRAM ***                                              01740010
  310 WRITE(6,320) AA(KNPP1),AA(JNPP1)                                 01750019
  320 FORMAT(5X,'NPP1 FILE NOT SORTED BY FSCM IN ASCENDING ORDER  ',A5, 01760010
     15X,A5)                                                           01770010
      GO TO 190                                                        01780036
  330 KNPP1=JNPP1                                                      01790021
      DO 331 N=1,JNPP1                                                 01800021
       WRITE(4,332) DUM1(N),AA(N),DUM2(N),BB(N),CC(N),DUM3(N),NOAR(N)  01810012
     1 ,AOPER(N),WMDR(N),DAYSCL(N),N,KNPP1                             01820035
  332  FORMAT(A10,A5,A4,I2,I2,A305,I2,A2,F5.2,I4,2I3)                  01830035
       ICOUNT=ICOUNT+1                                                 01840016
  331 CONTINUE                                                         01850011
      WRITE(6,333) IFSCM,IYR,IMONTH,ICOUNT                             01860016
  333 FORMAT(5X,4I10)                                                  01870016
  335 STOP                                                             01880017
      END                                                              01890000
```

Appendix E

The QUEST Model FORTRAN Source Code and JCL

Program Control File

A four record disk file is accessed by QUEST to provide the following parameters. This file is updated by the computer operator prior to execution of QUEST. The program code addressing this file appears on pages E-5 and E-6.

First Record - Specifies Start Date.

Position 1 - 2; Month to begin Model measurement - 2N.

Position 3 - 4; Year to begin Model measurement - 2N.

Second Record - Specifies End Date.

Position 1 - 2; Month to end Model measurement - 2N.

Position 3 - 4; Year to end Model measurement - 2N.

Third Record

Position 1 - 2; Minimum number of monthly data points before the start month needed to have sufficient history to compute effectiveness scores - 2N. Throughout the model development and validation phases of this effort, this parameter was set at 5 months.

Position 3 - 4; Year before which all historical data is ignored - 2N.

Fourth Record

Position 1; Specifies whether or not the Alert File will be used by the model. If the Alert File is unavailable or not desired, position is "1".

Position 2; Specifies the output products. A "0" or blank will suppress all output files except a hard copy printout of the final model results. A "1" will cause intermediate disk output files to be created at various stages of the model, enabling more detailed analysis and diagnosis, if desired.

Record Selection Process

Records are screened by QUEST. Not all facilities or monthly records are evaluated from the merged tape file addressed in paragraph IIB.

Inactivity. If no QAR hours were reported against an FSCM in a given month, the monthly record is deleted (see Page E-6). If there are fewer months of available history than specified in the program control file (see Page E-2), the total history for the FSCM is deleted (see also Page E-6). Furthermore, if QUEST is asked to measure effectiveness over a longer period of time than the historical data will allow, a message is generated and the available monthly records are moved to the disk file but are disregarded in subsequent steps (see Page E-15 and E-17).

Invalid or Missing Data. The model checks certain key elements per Page E-6.

Date of Data. If the data is older than specified by the Program Control File, the record is deleted. If the data is newer than the end date of the model it is likewise deleted (see Page E-6).

```
//GOR6040F JOB (6040,GOR),'GROVER',CLASS=3,MSGCLASS=V                   00010099
//RUNFTN EXEC FORTVCG,FVREGN=2500K,GOREGN=2000K,                        00020099
//     PARM.GO='LET,NORES,EP=MAIN,SIZE=500000'                          00021099
//FORT.SYSIN DD DSN=GOR.GROVER.FOR(MODEL6),DISP=SHR                     00040099
//GO.FT01F001 DD DSN=GOR.GROVER.ATL.INTAPE.JUL87,DISP=SHR               00050099
//GO.FT02F001 DD DSN=GOR.GROVER.ATL.JUL87.DISKYN,UNIT=3380,             00062099
//       DISP=(NEW,CATLG,DELETE),VOL=SER=WORKWS,                        00063099
//       DCB=(LRECL=227,RECFM=FB,BLKSIZE=11350),                        00064099
//       SPACE=(CYL,(6,4),RLSE)                                         00065099
//GO.FT03F001 DD DSN=GOR.GROVER.FOR(TESTCTL),DISP=SHR                   00070099
//GO.FT04F001 DD DSN=GOR.GROVER.ATLLBL.YN,UNIT=3380,                    00081099
//       DISP=(NEW,CATLG,DELETE),VOL=SER=WORKWS,                        00082099
//       DCB=(LRECL=239,RECFM=FB,BLKSIZE=11950),                        00083099
//       SPACE=(CYL,(6,1),RLSE)                                         00084099
//GO.FT09F001 DD DSN=GOR.GROVER.ATL.TOPYN,UNIT=3380,                    00085099
//       DISP=(NEW,CATLG,DELETE),VOL=SER=WORKWS,                        00086099
//       DCB=(LRECL=172,RECFM=FB,BLKSIZE=17200),                        00087099
//       SPACE=(CYL,(3,1),RLSE)                                         00088099
//GO.FT11F001 DD DSN=GOR.GROVER.ATL.OUTYN,UNIT=3380,                    00089099
//       DISP=(NEW,CATLG,DELETE),VOL=SER=WORKWS,                        00089199
//       DCB=(LRECL=122,RECFM=FB,BLKSIZE=12200),                        00089299
//       SPACE=(CYL,(2,1),RLSE)                                         00089399
//GO.FT08F001 DD DSN=GOR.GROVER.ALERT.DAUG87,DISP=SHR                   00089499
//GO.FT10F001 DD DSN=GOR.GROVER.JUL.MASTER.LONG,DISP=SHR                00089599
//GO.FT06F001 DD SYSOUT=*                                               00320099
//SYSOUT DD SYSOUT=*                                                    00330099
//SYSUDUMP DD SYSOUT=*                                                  00340099
//SYSPRINT DD SYSOUT=*                                                  00350099
//*                                                                     00360099
//*                                                                     00370099
//*  ***************************************************              00380099
//*  * THIS PROGRAM WILL SORT ON DATE AND QAORG CODE     *             00390099
//*  * AND TYPE AND EFFECTIVENESS.                       *             00400099
//*  ***************************************************              00410099
//*                                                                     00420099
//*                                                                     00430099
//*    ***************************************                          00440099
//*    THIS SEGMENT SORTS GOR.GROVER.ATL.OUTYN *                        00450099
//*    ***************************************                          00460099
//*                                                                     00470099
//STEP1 EXEC PGM=IE  'C000,PARM='MSG=AP',REGION=2000K                   00480099
//SORTLIB  DD DSN=S\   '.SORTLIB,DISP=SHR                               00490099
//SYSOUT   DD SYSOUT                                                    00500099
//SYSPRINT DD SYSOUT=*                                                  00510099
//SORTIN   DD DSN=GOR.\  OVER.ATL.OUTYN,DISP=SHR                        00520099
//*                                                                     00530099
//SORTOUT  DD DSN=GOR.GROVER.ATL.REPYN,DISP=(NEW,CATLG,DELETE),         00540099
//       SPACE=(CYL,(2,1),RLSE),UNIT=3380,VOL=SER=WORKWS                00550099
//SORTWK01 DD UNIT=WORKD,SPACE=(CYL,10)                                 00560099
//SORTWK02 DD UNIT=WORKD,SPACE=(CYL,10)                                 00570099
//SORTWK03 DD UNIT=WORKD,SPACE=(CYL,10)                                 00580099
//SORTWK04 DD UNIT=WORKD,SPACE=(CYL,10)                                 00590099
//SORTWK05 DD UNIT=WORKD,SPACE=(CYL,10)                                 00600099
//SYSIN    DD *                                                         00610099
 SORT FIELDS=(14,3,CH,A,11,3,CH,A,1,3,CH,A,10,1,CH,D,111,5,CH,A)        00620099
 END                                                                    00630099
//                                                                      00640099
```

```
C     *** THIS PROGRAM IS THE VERSION  USED FOR FINAL REPORT DOCUMENT    00010015
C     ***THIS MODULE IS RUN AFTER A MASTER TAPE HAS BEEN CREATED        00020002
C     BY MERGING THE MIS WITH THE MDR AND FACILITY PROFILE.***          00030002
C                                                                       00040002
C                                                                       00050002
C                                                                       00060002
C     DECLARE AND ARRAY VARIABLES                                       00070002
C                                                                       00080002
      CHARACTER DCASR(120)*6,QAORG(120)*3,FSCM(120)*6,TYP(120)*1,       00090002
     1CMDTY(120)*2,PVN(120)*1,ADCASR*6,BFSCM(2000)*5,OPER(120)*2,       00100002
     2FLAG(120)*18,DLRIN(120)*12,DLROUT(120)*12,DLROH(120)*12           00110002
      INTEGER MONTH(120),YEAR(120),PLANHR(120),LOTINS(120),PVIHR(120),  00120002
     1PEHR(120),PRHR(120),AQDR(120),BQDR(120),CQDR(120),DQDR(120),      00130002
     2EQDR(120),QDRHR(120),TVLHR(120),TNGHR(120),FMSHR(120),ADMNHR(120),00140002
     3SHIPMT(120),WD(120),WDHR(120),INTHR(120),REINHR(120),VISIT(120),  00150002
     4MRBHR(120),ECP(120),ECPHR(120),MDRHR(120),MTGHR(120),MRB(120),    00160002
     5POHR(120),PCO(120),CAQ(120),CONTR(120),DEGREE(120),JCIP(120),     00170002
     6ACNTRT(120),BCNTRT(120),QCNTRT(120),ISTRAT(120),                  00180002
     7QALIIN(120),QALIRE(120),ADNHND(120),BONHND(120),QONHND(120),      00190002
     8NQDR(120),PVINP(120),PELONP(120),PEELNP(120),NQAR(120),           00200002
     9ISEQ(120),RECS(120),DAYSCL(120),STARTM,STARTY,ENDMO,ENDYR         00210002
      REAL SI(120),EPA(120),ALRR(120),TOPCA(120),CARATE(120),           00220002
     1WDRATE(120),ECPRAT(120),AMRBRA(120),EPARAT(120),ALRRRA(120),      00230002
     2WMDRRA(120),TOPEPA(120),TOPMRB(120),TOPWD(120),TOPLRR(120),       00240002
     3TOPECP(120),WMDR(120),TOPMDR(120),A(628,36),AIDEAL(4,7),          00250002
     4AWORST(4,7),BIDEAL(4,7),BWORST(4,7),WEIGHT(4,7),TOPSCR(12,120)    00260002
      COMMON /RVAR/A                                                    00270002
      COMMON /CHTR/BFSCM,FLAG                                           00280002
C                                                                       00290002
C     ***THIS MODULE SELECTS RELEVANT DATA RECORDS FROM THE INPUT FILE. 00300002
C     PERFORMS EDIT CHECKS AND WRITES THE RECORD TO A DISK FILE.***     00310002
C                                                                       00320002
C     ***STEP 1 ***                                                     00330002
C     ***INITIALIZE PARAMETERS                                          00340002
C                                                                       00350002
      STARTM=99                                                         00360002
      STARTY=99                                                         00370002
      ENDMO=99                                                          00380002
      ENDYR=99                                                          00390002
      MONTHS=12                                                         00400002
C     ***EACH DCASR CODE SHOULD BE INSERTED NEXT; CREATING A UNIQUE PGM 00410002
C     ***OMIT NEXT LINE FOR DLA-WIDE MODEL****                          00420002
C     ADCASR='S1102A'                                                   00430002
      LCOUNT=0                                                          00440002
      NCOUNT=0                                                          00450002
      KCOUNT=0                                                          00460002
      JERROR=0                                                          00470002
C                                                                       00480002
C     ***ENTER TIME FRAME LIMITS ***                                    00490002
C     AN EXTERNAL FILE MUST BE SET UP TO CONTAIN DATES TO CONTROL MODEL 00500002
C     STARTM AND STARTY ARE THE MONTH AND YEAR TO BEGIN MEASURING QA EFF00510002
C     ENDMO AND ENDYR ARE THE MONTH AND YEAR THE MODEL STOPS            00520002
C     MONTHS ARE THE MINIMUM NUMBER OF DATA POINTS OF HISTORY NEEDED    00530002
C     CUTYR DELETES ALL DATA PRIOR TO CY SPECIFIED                      00540002
C     LONG INDICATES WHETHER OR NOT BACKUP FILES ARE CREATED.           00550002
C     ICIP SPECIFIES WHETHER OR NOT ALERT FILE IS USED.                 00560002
C                                                                       00570002
      READ(3,10) STARTM,STARTY                                          00580002
C     PERFORM EDIT CHECKS ON DATES ENTERED                              00590002
      CALL TIMCHK(STARTM,STARTY,JERROR)                                 00600002
      IF(JERROR .EQ. 1) GOTO 30                                         00610002
      READ(3,10) ENDMO,ENDYR                                            00620002
C     PERFORM EDIT CHECKS ON DATES ENTERED                              00630002
      IF(STARTY .GT. ENDYR) GOTO 30
```

```
      CALL TIMCHK(ENDMO,ENDYR,JERROR)                                   00640002
      IF(JERROR.EQ.1) GOTO 30                                           00650002
      READ(3,10) MONTHS,ICUTYR                                          00660002
      IF(MONTHS .LT. 3) GOTO 20                                         00670002
      IF(MONTHS .GT. 99) GOTO 20                                        00680002
      READ(3,11) LONG,ICIP                                              00690002
      WRITE(6,16) STARTM,STARTY,ENDMO,ENDYR                             00700002
      GO TO 100                                                         00710002
   10 FORMAT(2I2)                                                       00720002
   11 FORMAT(2I1)                                                       00730002
   16 FORMAT(1X,'INPUT ACCEPTED NORMALLY. NO DEFAULTS. START',2I3,2X,   00740002
     1'END',2I3)                                                        00750002
   20 WRITE(6,25) MONTHS                                                00760002
   25 FORMAT(1X,'HISTORY OUTSIDE A_LOWABLE RANGE',I2)                   00770002
      GO TO 202                                                         00780002
   30 WRITE(6,35) STARTM,STARTY,ENDMO,ENDYR                             00790002
   35 FORMAT(1X,'DATES NOT ACCEPTED, TRY AGAIN',4I4)                    00800002
      GO TO 202                                                         00810002
C                                                                       00820002
C     *** STEP 2****                                                    00830002
C     ***READ A RECORD FROM MASTER DATA FILE ***                        00840002
C     ***CHECK FOR MISSING OR INVALID DATA ***                          00850002
C                                                                       00860002
  100 DO 195 I=1,120                                                    00870002
       READ(1,101,END=202) DCASR(I),QAORG(I),FSCM(I),TYP(I),            00880002
     1CMDTY(I),PVN(I),MONTH(I),YEAR(I),PLANHR(I),LOTINS(I),             00890002
     2PVIHR(I),PEHR(I),PRHR(I),AQDR(I),BQDR(I),CQDR(I),DQDR(I),         00900002
     3EQDR(I),QDRHR(I),TVLHR(I),TNGHR(I),FMSHR(I),ADMNHR(I),SHIPMT(I), 00910002
     4WD(I),WDHR(I),INTHR(I),REINHR(I),VISIT(I),MRBHR(I),ECP(I),        00920002
     5ECPHR(I),MDRHR(I),MTGHR(I),MRB(I),POHR(I),PCO(I),CAO(I),SI(I),    00930002
     6EPA(I),ALRR(I),CONTR(I),DLRIN(I),DLROUT(I),ACNTRT(I),BCNTRT(I),   00940002
     7OCNTRT(I),QALIIN(I),QALIRE(I),AONHNO(I),BONHNO(I),OONHNO(I),       00950002
     8DLROH(I),NQDR(I),PVINP(I),PELONP(I),PEELNP(I),NQAR(I),OPER(I),    00960002
     9WMDR(I),DAYSCL(I),ISEQ(I),RECS(I)                                 00970002
  101 FORMAT(A6,A3,A6,A1,A2,A1,2I2,I4,I4,4X,10X,2(I4,3X),I4,I3,2I2,2I1, 00980002
     13I4,14X,3I4,I2,I4,2I3,I2,I4,I3,2I4,2X,I4,3X,I4,I3,2I2,10X,3F5.2,  00990002
     25X,I6,6X,2A12,5I6,4X,3I6,A12,I2,19X,I5,I3,I5,5X,I2,A2,F5.2,I4,2I3)01000002
       KCOUNT=KCOUNT+1                                                  01010002
       ISTRAT(I)=0                                                      01020002
C      DELETE NEXT LINE FROM DLA-WIDE MODEL                             01030002
C      IF(DCASR(I) .NE. ADCASR) GOTO 190                                01040002
       IF(QAORG(I) .EQ. '    ') GOTO 190                                01050002
       IF(FSCM(I)  .EQ. '     ') GOTO 190                               01060002
       IF((TYP(I) .NE. 'N') .AND. (TYP(I) .NE. 'R')) GOTO 190           01070002
       IF(CMDTY(I) .GT. 'ZZ') GOTO 190                                  01080002
       IF((PVN(I) .NE. 'A') .AND. (PVN(I) .NE. 'B') .AND. (PVN(I) .NE.  01090002
     1'C')) GOTO 190                                                    01100002
       IF(RECS(I) .LT. MONTHS) GOTO 190                                 01110002
       IF(YEAR(I) .GT. ENDYR) GOTO 190                                  01120002
       IF((YEAR(I) .EQ. ENDYR) .AND. (MONTH(I) .GT. ENDMO)) GOTO 190    01130002
       IF (YEAR(I) .LT. ICUTYR) GOTO 190                                01140002
       ITOTHR=PLANHR(I)+PVIHR(I)+PEHR(I)+PRHR(I)+QDRHR(I)+TVLHR(I)+     01150002
     1TNGHR(I)+FMSHR(I)+ADMNHR(I)+WDHR(I)+INTHR(I)+REINHR(I)+MRBHR(I)+  01160002
     2ECPHR(I)+MDRHR(I)+MTGHR(I)+POHR(I)                                01170002
       IF((ITOTHR .LE. 0) .AND. (WMDR(I) .EQ. 0.0)) GOTO 190            01180002
C      IF RECORD SURVIVES EDIT CHECK, ASSIGN A STRAT ID NUMBER          01190002
       LCOUNT=LCOUNT + 1                                                01200002
       CALL STRAT(I,TYP,CMDTY,PVN,OPER,NQAR,ISTRAT)                     01210002
C      WRITE NEW RECORD TO DISK FOR FURTHER USE                         01220002
       WRITE(2,150) DCASR(I),QAORG(I),FSCM(I),TYP(I),CMDTY(I),PVN(I),MONO1230002
     1TH(I),YEAR(I),PLANHR(I),LOTINS(I),PVIHR(I),                       01240002
     2PEHR(I),AQDR(I),BQDR(I),CQDR(I),                                  01250002
     3DQDR(I),EQDR(I),ADMNHR(I),SHIPMT(I),WD(I),INTHR(I),REINHR(I),VISITO1260002
     4(I),ECP(I),MTGHR(I),MRB(I),PCO(I),CAO(I),SI(I),EPA(I),ALRR(I),    01270002
     5CONTR(I),DLRIN(I),DLROUT(I),ACNTRT(I),BCNTRT(I),OCNTRT(I),QALIIN  01280002
     6(I),QALIRE(I),AONHNO(I),BONHNO(I),OONHNO(I),DLROH(I),NQDR(I),      01290002
```

```
     7PVINP(I),PELONP(I),PEELNP(I),NOAR(I),OPER(I),WMOR(I),DAYSCL(I),    01300002
                                                                         01310002
     8ISEQ(I),RECS(I),ISTRAT(I)                                          01320002
  150  FORMAT(A6,A3,A6,A1,A2,A1,2I2,4I4,I3,2I2,2I1,2I4,I2,2I3,I2,I3,2I4,01330002
     1212,3F6.2,I6,2A12,8I6,A12,I2,I5,I3,I5,I2,A2,F5.2,I4,2I3,I4)        01340002
       GO TO 195                                                         01350002
  190  NCOUNT=NCOUNT+1                                                   01360002
  195 CONTINUE                                                           01370002
      GO TO 100                                                          01380002
C                                                                        01390002
C     EXIT MODULE                                                        01400002
C                                                                        01410002
  202 WRITE(6,203) KCOUNT,LCOUNT,NCOUNT                                  01420002
  203 FORMAT(5X,'RECORDS READ',1X,I9,10X,'RECORDS WRITTEN ',I9,10X,'RECO01430002
     1RDS SKIPPED ',I9)                                                  01440002
C                                                                        01450002
C*************************************************************         01460002
C     ***THIS SECTION SELECTS DATA RECORDS FROM THE INPUT DISK FILE,      01470014
C     COMPARES FSCM WITH CONTRACTOR ALERT FILE, AND COMPUTES             01480014
C     AN INDEX BASED ON ALERT, MDRS AND SEVERE QDRS.                     01490002
C     CODES ASSIGNED ARE 1-4 AS FOLLOWS: 1 IS A PROBLEM RESIDENT         01500002
C     FACILITY, 2 IS A NORMAL RESIDENT, 3 IS A PROBLEM NONRESIDENTAND    01510002
C     4 IS A NORMAL NONRESIDENT.                                         01520002
C     ***STEP 1 ***                                                      01530002
C     ***INITIALIZE PARAMETERS                                           01540002
C                                                                        01550002
      REWIND 2                                                           01560002
      IFAC=0                                                             01570002
      JFAC=0                                                             01580002
      ICIPNO=0                                                           01590002
C                                                                        01600002
C  READ THE MASTER DATA FILE                                             01610002
C                                                                        01620002
      DO 640 I=1,628                                                     01630002
      READ (10,641) A(I,1),A(I,2),A(I,3),A(I,4),A(I,5),A(I,6),A(I,7),    01640002
     1A(I,8),A(I,9),A(I,10),A(I,11),A(I,12),A(I,13),A(I,14),A(I,15),     01650002
     2A(I,16),A(I,17),A(I,18),A(I,19),A(I,20),A(I,21),A(I,22),A(I,23),   01660002
     3A(I,24),A(I,25),A(I,26),A(I,27),A(I,28),A(I,29),A(I,30),A(I,31),   01670002
     4A(I,32),A(I,33),A(I,34),A(I,35),A(I,36)                            01680002
  641 FORMAT(3X,36F7.2)                                                  01690002
  640 CONTINUE                                                           01700002
C                                                                        01710002
C  SET THE IDEAL AND NEGATIVE IDEAL CONDITIONS AND WEIGHT FACTORS        01720002
      CALL CORNER(AIDEAL,AWORST,BIDEAL,BWORST,WEIGHT)                    01730002
C                                                                        01740002
C  CHECK TO SEE IF CONTRACTOR ALERT OPTION IS ACTIVE                     01750002
C                                                                        01760002
      IF (ICIP .EQ. 1) GOTO 674                                          01770002
C     READ THE CONTRACTOR ALERT FILE                                     01780002
      DO 671 I=1,2000                                                    01790002
       ICIPNO=ICIPNO+1                                                   01800002
       READ(8,672,END=673) BFSCM(I)                                      01810002
  672  FORMAT(4X,A5)                                                     01820002
  671 CONTINUE                                                           01830002
C     IF THIS LOOP IS DONE THE ENTIRE FILE EXCEEDS THE ARRAY LENGTH      01840002
      GO TO 675                                                          01850002
  673 ICIPNO=I-1                                                         01860002
C     SET DEFAULTS TO ZERO                                               01870002
  674 DO 670 I=1,120                                                     01880002
      JCIP(I)=0                                                          01890002
      DEGREE(I)=0                                                        01900002
  670 CONTINUE                                                           01910002
C     ***READ FIRST RECORD FOR FACILITY***                               01920002
  675  READ(2,201,END=999) DSADR(I),CAGE(I),FSCM(I),TYP(I),SMDTY(I),      01930002
     1PVN(I),MONTH(I),YEAR(I),PLANHR(I),LOTINS(I),PVIHR(I),              01940002
     2PEHR(I),AQDR(I),                                                   01950002
     3BQDR(I),CQDR(I),DQDR(I),EQDR(I),ADMNHR(I),SHIPMT(I),
```

```
     4WD(1),INTHR(1),REINHR(1),VISIT(1),ECP(1),MTGHR(1),MRB(1),PCO(1),   01960002
     5CAO(1),SI(1),EPA(1),ALRR(1),CONTR(1),DLRIN(1),DLROUT(1),ACNTRT(1),01970002
     6BCNTRT(1),OCNTRT(1),QALIIN(1),QALIRE(1),AONHND(1),BONHND(1),       01980002
     7OONHND(1),DLROH(1),NQDR(1),PVINP(1),PELONP(1),PEELNP(1),NOAR(1),   01990002
     8OPER(1),WMDR(1),DAYSCL(1),ISEQ(1),RECS(1),ISTRAT(1)                02000002
  601 FORMAT(A6,A3,A6,A1,A2,A1,2I2,4I4,I3,2I2,2I1,2I4,I2,2I3,I2,I3,2I4,  02010002
     1212,3F6.2,I6,2A12,8I6,A12,I2,I5,I3,I5,I2,A2,F5.2,I4,2I3,I4)        02020002
C      JFAC COUNTS THE FACILITIES THAT REACH THIS STEP.KB COUNTS THE      02030002
C      NUMBER OF RECORDS FOR A GIVEN FACILITY                            02040002
       JFAC=JFAC+1                                                       02050002
       KB=1                                                              02060002
       IF(ICIP .EQ. 1) GOTO 604                                          02070002
C      IF THE ALERT OPTION IS ACTIVE, CHECK TO SEE IF FSCM IS ON ALERT   02080002
       CALL CIP(KB,FSCM,JCIP,ICIPNO)                                     02090002
C      READ ADDITIONAL RECORDS FOR THE FSCM                              02100002
  604 DO 650 KB=2,120                                                    02110002
       READ(2,601,END=666) DCASR(KB),OAORG(KB),FSCM(KB),TYP(KB),CMDTY(KB)02120002
     1),PVN(KB),MONTH(KB),YEAR(KB),PLANHR(KB),LOTINS(KB),PVIHR(KB),      02130002
     2PEHR(KB),                                                          02140002
     3AQDR(KB),BQDR(KB),CQDR(KB),DQDR(KB),EQDR(KB),ADMNHR(KB),SHIPMT(KB)02150002
     4,WD(KB),INTHR(KB),REINHR(KB),VISIT(KB),ECP(KB),                    02160002
     5MTGHR(KB),MRB(KB),PCO(KB),CAO(KB),SI(KB),EPA(KB),ALRR(KB),         02170002
     6CONTR(KB),DLRIN(KB),DLROUT(KB),ACNTRT(KB),BCNTRT(KB),OCNTRT(KB),   02180002
     7QALIIN(KB),QALIRE(KB),AONHND(KB),BONHND(KB),OONHND(KB),DLROH(KB),  02190002
     8NQDR(KB),PVINP(KB),PELONP(KB),PEELNP(KB),NOAR(KB),OPER(KB),        02200002
     9WMDR(KB),DAYSCL(KB),ISEQ(KB),RECS(KB),ISTRAT(KB)                   02210002
       KA=KB-1                                                           02220002
       JCIP(KB)=JCIP(1)                                                  02230002
C      STOP READING IF THIS IS THE LAST RECORD FOR FSCM OR A NEW FSCM     02240002
C      APPEARS                                                           02250002
       IF (ISEQ(KB) .EQ. RECS(KB)) GOTO 652                              02260002
       IF (FSCM(KB) .GT. FSCM(KA)) GOTO 651                              02270002
  650 CONTINUE                                                           02280002
       GO TO 678                                                         02290002
  651 NCTL=1                                                             02300002
       GO TO 654                                                         02310002
  652 NCTL=0                                                             02320002
       KA=KA+1                                                           02330002
  654 IF(KA .LT. MONTHS) GOTO 662                                        02340002
C      IF ENOUGH DATA, ASSIGN DEGREE OF DIFFICULTY INDEX TO FSCM ARRAY.  02350002
C      THEN PREPARE ARRAY FOR TOPSIS BY COMPUTING INDICATORS.            02360002
       DO 660 L1=1,KA                                                    02370002
        CALL DIFF(TYP,CQDR,DQDR,EQDR,WMDR,L1,JCIP,DEGREE,ISTRAT)         02380002
        IF (LONG .EQ. 0) GOTO 660                                        02390002
        WRITE(4,661) DCASR(L1),OAORG(L1),FSCM(L1),TYP(L1),CMDTY(L1),     02400002
     1PVN(L1),MONTH(L1),YEAR(L1),PLANHR(L1),LOTINS(L1),PVIHR(L1),        02410002
     2PEHR(L1),AQDR(L1),BQDR(L1),CQDR(L1),DQDR(L1),EQDR(L1),ADMNHR(L1),  02420002
     3SHIPMT(L1),WD(L1),INTHR(L1),REINHR(L1),VISIT(L1),ECP(L1),          02430002
     4MTGHR(L1),MRB(L1),PCO(L1),CAO(L1),SI(L1),EPA(L1),ALRR(L1),         02440002
     5CONTR(L1),DLRIN(L1),DLROUT(L1),ACNTRT(L1),BCNTRT(L1),OCNTRT(L1),   02450002
     6QALIIN(L1),QALIRE(L1),AONHND(L1),BONHND(L1),OONHND(L1),DLROH(L1),  02460002
     7NQDR(L1),PVINP(L1),PELONP(L1),PEELNP(L1),NOAR(L1),OPER(L1),        02470002
     8WMDR(L1),DAYSCL(L1),ISEQ(L1),RECS(L1),ISTRAT(L1),L1,KA,JCIP(L1),   02480002
     9DEGREE(L1)                                                         02490002
  661 FORMAT(A6,A3,A6,A1,A2,A1,2I2,4I4,I3,2I2,2I1,2I4,I2,2I3,I2,I3,2I4,  02500002
     1212,3F6.2,I6,2A12,8I6,A12,I2,I5,I3,I5,I2,A2,F5.2,I4,2I3,I4,4I3)    02510002
  660 CONTINUE                                                           02520002
C      COUNT THE FACILITIES THAT HAVE SURVIVED.                          02530002
       IFAC=IFAC+1                                                       02540002
C      COMPUTE TOPSIS VALUES FOR FSCM                                    02550002
        CALL PREPIN(KA,AQDR,BQDR,CQDR,DQDR,EQDR,WD,ECP,MRB,EPA,ALRR,WMDR,02560002
     1TOPCA,CARATE,WDRATE,ECPRAT,AMRBRA,EPARAT,ALRRRA,WMDRRA,STARTM,     02570002
     2STARTY,ENDMO,ENDYR,FSCM,MONTH,YEAR,TOPWD,TOPECP,TOPMRB,TOPEPA,     02580002
     3TOPLRR,TOPMDR,DCASR,OAORG,DEGREE,AIDEAL,AWORST,BIDEAL,BWORST,      02590002
     4ISTRAT,WEIGHT,REINHR,INTHR,MTGHR,SHIPMT,LOTINS,VISIT,PVN,TYP,      02600002
     5PEHR,CONTR,PLANHR,DAYSCL,PCO,CAO,NQDR,SI,ADMNHR,PVINP,PELONP,      02610002
```

```
     6PEELNP,QALIIN,QALIRE,LONG,TOPSCR,PVIHR)                                  02620002
C      BEGIN ANOTHER FSCM.                                                     02630002
C      EITHER READ A NEW RECORD OR MOVE LAST RECORD INTO FIRST POSITION        02640002
  662  IF(NCTL .EQ. 0) GOTO 600                                                02650002
       CALL NEWFAC(KB,DCASR,QAORG,FSCM,TYP,CMDTY,PVN,MONTH,YEAR,PLANHR,        02660002
      1LOTINS,PEHR,AQDR,BQDR,CQDR,DQDR,EQDR,ADMNHR,SHIPMT,WD,INTHR,            02670002
      2REINHR,VISIT,ECP,MTGHR,MRB,PCO,CAO,SI,EPA,ALRR,CONTR,DLRIN,DLROUT,      02680002
      3ACNTRT,BCNTRT,CCNTRT,QALIIN,QALIRE,AONHND,BONHND,CONHND,DLROH,          02690002
      4NQDR,PVINP,PELONP,PEELNP,NQAR,QPER,WMDR,DAYSCL,ISEQ,RECS,ISTRAT,        02700002
      5JCIP,ICIPNO,ICIP,PVIHR)                                                 02710002
       JFAC=JFAC+1                                                             02720002
       GO TO 604                                                               02730002
C      ABNORMAL TERMINATION                                                    02740002
  675  WRITE(6,676) ICIPNO                                                     02750002
  676  FORMAT(5X,'EXCESS RECORDS ON CONTRACTOR INPROVEMENT FILE,OVER',I6)      02760002
  678  WRITE(6,679)                                                            02770002
  679  FORMAT(2X,'ERROR DETECTED. VERIFY INPUT FILE SORTED BY FSCM')           02780002
C      NORMAL TERMINATION                                                      02790002
  666  WRITE(6,668) JFAC,IFAC                                                  02800002
  668  FORMAT(5X,'FACILITIES IN ',I9,5X,'FACILITIES OUT ',I9)                  02810002
       WRITE(6,677) ICIPNO                                                     02820002
  677  FORMAT(5X,'CIP FACILITIES',I6)                                          02830002
       STOP                                                                    02840002
       END                                                                     02850002
C                                                                              02860002
       SUBROUTINE TIMCHK(MONTH,IYEAR,JERROR)                                   02870002
C      CHECKS TIME VALUES ENTERED FOR INCONSISTENCY                            02880002
       IF(IYEAR .LT. 84) GOTO 300                                              02890002
       IF(MONTH .LT. 1) GOTO 300                                               02900002
       IF(MONTH .GT. 12) GOTO 300                                              02910002
       JERROR=0                                                                02920002
       GO TO 310                                                               02930002
  300  JERROR=1                                                                02940002
  310  RETURN                                                                  02950002
       END                                                                     02960002
C                                                                              02970002
       SUBROUTINE STRAT(I,TYP,CMDTY,PVN,OPER,NQAR,JJ)                          02980002
C                                                                              02990002
C      ASSIGNS A STRATIFICATION ID NUMBER TO EACH RECORD. STRAT ID             03000002
C      IS USED TO GROUP SIMILAR FACILITIES. THE NUMBER RANGES INITIALLY        03010002
C      FROM 1-628. ID VALUES OF 1-240 ARE USED FOR RESIDENT FACILITIES,        03020002
C      NON-RESIDENT ARE 241-624. REPAIR/OVERHAUL FACILITIES ARE 626,627,       03030002
C      AND 628 FOR NONRESIDENT, SMALL RESIDENT AND LARGE RESIDENT RESP.        03040002
C      NONASSIGNED FACILITIES ARE PUT IN GROUP 625.                            03050002
C                                                                              03060002
       CHARACTER TYP(120)*1,CMDTY(120)*2,PVN(120)*1,OPER(120)*2                03070002
       INTEGER NQAR(120),JJ(120)                                               03080002
C                                                                              03090002
       IF((OPER(I) .EQ. 'C ') .OR. (OPER(I) .EQ. ' C') .OR. (CMDTY(I)          03100002
      1.EQ. 'A5')) GOTO 598                                                    03110002
C      KK IS A DORO FORTRAN UNIQUE VALUE OF 1-16 REPRESENTING THE              03120002
C      16 COMMODITY CODES IN DLAM 8200.2                                       03130002
       KK=(ICHAR(CMDTY(I)(1:1)))-192                                           03140002
         IF(KK .GT. 33) GOTO 501                                               03150002
         IF(KK .GT. 9) GOTO 502                                                03160002
         GO TO 504                                                             03170002
  501    KK=KK-15                                                              03180002
         GO TO 504                                                             03190002
  502    KK=KK-7                                                               03200002
  504    IF (KK .EQ. 16) GOTO 515                                              03210002
         IF (KK .GT. 21) GOTO 516                                              03220002
         IF (KK .GT. 18) GOTO 517                                              03230002
         IF (KK .GT. 10) GOTO 518                                              03240002
         IF (KK .GT. 6) GOTO 519                                               03250002
         GO TO 520                                                             03260002
  515    KK=12                                                                 03270002
```

```
                                                                      03280002
      GO TO 520                                                       03290002
  516 KK=KK-8                                                         03300002
      GO TO 520                                                       03310002
  517 KK=KK-6                                                         03320002
      GO TO 520                                                       03330002
  518 KK=KK-3                                                         03340002
      GO TO 520                                                       03350002
  519 KK=KK-1                                                         03360002
  520 IF (KK .GT. 16) GOTO 599                                        03370002
C     SPLIT RESIDENT FROM NONRESIDENT                                 03380002
      IF(TYP(I) .EQ. 'N') GOTO 550                                    03390002
C     RESIDENT ID VALUES DEPEND ON COMMODITY(16), QA PROV(3) AND      03400002
C     NUMBER OF QARS(4)                                               03410002
C COMBINE COMMODITIES                                                 03420002
      CALL STRAT1(KK)                                                 03430002
       KKK=KK*15-15                                                   03440002
C     QA PROV IS EITHER A OR B OR C (MIL Q, MIL I OR OTHER)           03450002
        IF (PVN(I) .EQ. 'A') GOTO 526                                 03460002
        IF (PVN(I) .EQ. 'B') GOTO 528                                 03470002
         LL=10                                                        03480002
         GO TO 530                                                    03490002
  526    LL=0                                                        03500002
         GO TO 530                                                    03510002
  528    LL=5                                                        03520002
  530   IF (NQAR(I) .LE. 2) GOTO 532                                  03530002
        IF (NQAR(I) .LE. 7) GOTO 534                                  03540002
        IF (NQAR(I) .LE. 20) GOTO 536                                 03550002
         MM=5                                                         03560002
         GO TO 540                                                    03570002
  532    MM=1                                                        03580002
         GO TO 540                                                    03590002
  534    MM=2                                                        03600002
         GO TO 540                                                    03610002
  536    MM=3                                                        03620002
         GO TO 540                                                    03630002
  540    JJ(I)=KKK+LL+MM                                             03640002
         GO TO 599                                                    03650002
C ASSIGN STRAT ID TO NONRESIDENT FACILITY                             03660002
C     VALUES RANGE FROM 241-624 DEPENDING ON 2 ALPHA COMMODITY CODE   03670002
C     IN DLAM 8200.2(16*8) AND QA PROV(3)                             03680002
  550   MMM=KK*24-24+240                                              03690002
        NN=0                                                          03700002
        IF(PVN(I) .EQ. 'A') GOTO 552                                  03710002
        IF((PVN(I) .EQ. 'B') .AND. ((KK .EQ. 7) .OR. (KK .EQ. 11) .OR. (03720002
     1 KK .EQ. 13))) GOTO 554                                         03730002
        IF((PVN(I) .EQ. 'C') .AND. (KK .EQ. 11)) GOTO 551             03740002
        NN=(ICHAR(CMDTY(I)(2:2))-240)*3-3                             03750002
        IF(PVN(I) .EQ. 'B') GOTO 554                                  03760002
  551   NNN=3                                                         03770002
        GO TO 556                                                     03780002
  552   NNN=1                                                         03790002
        GO TO 556                                                     03800002
  554   NNN=2                                                         03810002
  556   JJ(I)=MMM+NN+NNN                                              03820002
        IF((JJ(I) .GE. 253) .AND. (JJ(I) .LE. 255)) GOTO 598          03830002
        IF((JJ(I) .GE. 316) .AND. (JJ(I) .LE. 318)) GOTO 598          03840002
C COMBINE DIFFERENT STRATA                                            03850002
        CALL STRAT2(I,JJ)                                             03860002
        GO TO 599                                                     03870002
C ASSIGN STRAT ID TO MAINTENANCE FACILITIES                           03880002
  598   JJ(I)=626                                                     03890002
        IF((TYP(I) .EQ. 'R') .AND. (NQAR(I) .GE. 8)) JJ(I)=628        03900002
        IF((TYP(I) .EQ. 'R') .AND. (NQAR(I) .LE. 8)) JJ(I)=627        03910002
  599   IF(JJ(I) .EQ. 0) JJ(I)=625                                    03920002
        RETURN                                                        03930002
        END
```

```
      SUBROUTINE STRAT1(KK)                                              03940002
C   THIS SUBROUTINE COMBINES RESIDENT STRATA                             03950002
C     COMBINE MISSILES AND SPACE WITH AIRCRAFT                           03960002
        IF (KK .EQ. 16) THEN                                             03970002
          KK=1                                                           03980002
          RETURN                                                         03990002
        ENDIF                                                            04000002
C     COMBINE WEAPONS WITH MUNITIONS                                     04010002
        IF (KK .EQ. 15) THEN                                             04020002
          KK=2                                                           04030002
          RETURN                                                         04040002
        ENDIF                                                            04050002
C     COMBINE SERVICE WITH GENERAL                                       04060002
        IF (KK .EQ. 13) THEN                                             04070002
          KK=6                                                           04080002
          RETURN                                                         04090002
        ENDIF                                                            04100002
C     COMBINE ELCTRONIC SYSTEMS AND ELECTRICAL WITH ELECTRONIC           04110002
        IF ((KK .EQ. 8) .OR. (KK .EQ. 5)) THEN                           04120002
          KK=9                                                           04130002
          RETURN                                                         04140002
        ENDIF                                                            04150002
C     COMBINE CHEMICAL WITH PETROLEUM                                    04160002
        IF (KK .EQ. 7) THEN                                              04170002
          KK=12                                                          04180002
          RETURN                                                         04190002
        ENDIF                                                            04200002
C     COMBINE MARINE WITH VEHICLES                                       04210002
        IF (KK .EQ. 4) THEN                                              04220002
          KK=14                                                          04230002
          RETURN                                                         04240002
        ENDIF                                                            04250002
        RETURN                                                           04260002
        END                                                              04270002
      SUBROUTINE STRAT2(I,JJ)                                            04280002
C   THIS SUBROUTINE COMBINES NONRESIDENT STRATA                          04290002
      INTEGER JJ(120)                                                    043000C2
      IF((JJ(I).EQ.269).OR.(JJ(I).EQ.270).OR.(JJ(I).EQ.390)) THEN        04310002
        JJ(I)=JJ(I)-3                                                    04320002
        RETURN                                                           04330002
      ENDIF                                                              04340002
      IF((JJ(I).EQ.593).OR.(JJ(I).EQ.272).OR.(JJ(I).EQ.275)) THEN        04350002
        JJ(I)=278                                                        04360002
        RETURN                                                           04370002
      ENDIF                                                              04380002
      IF((JJ(I).EQ.594).OR.(JJ(I).EQ.273).OR.(JJ(I).EQ.276)) THEN        04390002
        JJ(I)=279                                                        04400002
        RETURN                                                           04410002
      ENDIF                                                              04420002
      IF((JJ(I) .EQ. 296) .OR. (JJ(I) .EQ. 302)) THEN                    04430002
        JJ(I)=305                                                        04440002
        RETURN                                                           04450002
      ENDIF                                                              04460002
      IF((JJ(I) .EQ. 297) .OR. (JJ(I) .EQ. 303)) THEN                    04470002
        JJ(I)=306                                                        04480002
        RETURN                                                           04490002
      ENDIF                                                              04500002
      IF((JJ(I) .EQ. 434) .OR. (JJ(I) .EQ. 435)) THEN                    04510002
        JJ(I)=JJ(I)+12                                                   04520002
        RETURN                                                           04530002
      ENDIF                                                              04540002
      IF((JJ(I).EQ.506).OR.(JJ(I).EQ.51[illegible].OR.(JJ(I).EQ.521)) THEN  04550002
        JJ(I)=524                                                        04560002
        RETURN                                                           04570002
      ENDIF                                                              04580002
      IF((JJ(I).EQ.507).OR.(JJ(I).EQ.516).OR.(JJ(I).EQ.522)) THEN        04590002
```

```
          JJ(I)=525                                                          04600002
          RETURN                                                             04610002
        ENDIF                                                                04620002
        IF((JJ(I) .EQ. 587) .OR. (JJ(I) .EQ. 590)) THEN                      04630002
          JJ(I)=581                                                          04640002
          RETURN                                                             04650002
        ENDIF                                                                04660002
        IF((JJ(I) .EQ. 588) .OR. (JJ(I) .EQ. 591)) THEN                      04670002
          JJ(I)=582                                                          04680002
          RETURN                                                             04690002
        ENDIF                                                                04700002
        IF((JJ(I) .EQ. 602) .OR. (JJ(I) .EQ. 605)) THEN                      04710002
          JJ(I)=251                                                          04720002
          RETURN                                                             04730002
        ENDIF                                                                04740002
        IF((JJ(I) .EQ. 603) .OR. (JJ(I) .EQ. 606)) THEN                      04750002
          JJ(I)=252                                                          04760002
          RETURN                                                             04770002
        ENDIF                                                                04780002
        IF((JJ(I) EQ.611) OR.(JJ(I).EQ.614).OR.(JJ(I) EQ.248)) THEN          04790002
          JJ(I)=260                                                          04800002
          RETURN                                                             04810002
        ENDIF                                                                04820002
        IF((JJ(I).EQ.612).OR.(JJ(I).EQ.615).OR.(JJ(I).EQ.249)) THEN          04830002
          JJ(I)=261                                                          04840002
          RETURN                                                             04850002
        ENDIF                                                                04860002
        IF((JJ(I) .EQ. 608) .OR. (JJ(I) .EQ. 609)) THEN                      04870002
          JJ(I)=JJ(I)-351                                                    04880002
          RETURN                                                             04890002
        ENDIF                                                                04900002
        IF((JJ(I) .EQ. 584) .OR. (JJ(I) .EQ. 585)) THEN                      04910002
          JJ(I)=JJ(I)-30                                                     04920008
          RETURN                                                             04930002
        ENDIF                                                                04940002
        IF((JJ(I) .EQ. 341) .OR. (JJ(I) .EQ. 342)) THEN                      04950002
          JJ(I)=JJ(I)+12                                                     04960002
          RETURN                                                             04970002
        ENDIF                                                                04980002
        RETURN                                                               04990002
        END                                                                  05000002
C                                                                            05010002
C *** THIS SUBROUTINE ASSIGNS A DIFFICULTY INDEX TO A FACILTY OF 1 TO 4.     05020002
      SUBROUTINE DIFF(TYP,CQDR,DQDR,EQDR,WMDR,L1,JCIP,DEGREE,ISTRAT)         05030002
      CHARACTER TYP(120)*1                                                   05040002
      INTEGER CQDR(120),DQDR(120),EQDR(120),JCIP(120),DEGREE(120),           05050002
     1ISTRAT(120)                                                            05060002
      REAL WMDR(120),AAVG(120),A(628,36)                                     05070002
      COMMON /RVAR/A                                                         05080002
      IF (TYP(L1) .EQ. 'R') GOTO 10                                          05090002
C     NONRESIDENT ASSIGNMENT IS 3 OR 4 DEPENDING ON ALERT MATCH, MDR         05100002
C     ACTIVITY OR CORRCTIVE ACTIONS C D E.EASY IS 4. HARD IS 3.              05110002
      DO 6 MM=1,L1                                                           05120002
        IF ((JCIP(L1) .EQ. 1) .OR.                                           05130002
     1((CQDR(L1)+DQDR(MM)+EQDR(L1)) .GT. 0)) GOTO 5                          05140002
        IF (MM .LT. L1-2) GOTO 6                                             05150002
        IF (WMDR(MM) .NE. 0.0) GOTO 5                                        05160002
    6 CONTINUE                                                               05170002
      DEGREE(L1)=4                                                           05180002
      GO TO 20                                                               05190002
    5 DEGREE(L1)=3                                                           05200002
      GO TO 20                                                               05210002
C     RESIDENT ASSIGNMENT IS 1 OR 2. EASY IS 2. HARD IS 1.                   05220002
   10 ASUM=0.0                                                               05230002
      AAVG(L1)=0.0                                                           05240002
      IBAD=0                                                                 05250002
```

```
      IF (JCIP(L1) .EQ. 1) IBAD=IBAD+1                                  05260C07
      DO 15 KK=1,L1                                                     05270002
       IF (DQDR(KK).GT. 0) IBAD=IBAD+1                                  05280002
       ASUM=ASUM+WMDR(KK)                                               05290002
       AAVG(KK)=ASUM/REAL(KK)                                           05300005
   15 CONTINUE                                                          05310002
      IF(AAVG(L1) .GT. (A(ISTRAT(L1),33)+A(ISTRAT(L1),34))) IBAD=IBAD+1 05320007
      IF((CQDR(L1)+EQDR(L1)) .GT. 0) IBAD=IBAD+1                        05330002
      DEGREE(L1)=2                                                      05340002
      IF(IBAD .GE. 2) DEGREE(L1)=1                                      05350002
   20 RETURN                                                            05360002
      END                                                               05370002
C *** THIS SUBROUTINE COMPARES THE FSCM OF RECORD WITH THE ALERT FILE.  05380C02
C     IF THERE IS A MATCH A VALUE OF 1 IS ASSIGNED TO JCIP,ELSE IT'S 0. 05390002
      SUBROUTINE CIP(J,AFSCM,KCIP,ICIPNO)                               05400002
      CHARACTER AFSCM(120)*6,BFSCM(2000)*5,CFSCM*5                      05410002
      INTEGER KCIP(120)                                                 05420002
      COMMON /CHTR/BFSCM,FLAG                                           05430002
      DO 10 I=1,ICIPNO                                                  05440002
       CFSCM=AFSCM(J)(2:6)                                              05450002
       IF(CFSCM .EQ. BFSCM(I)) GOTO 25                                  05460002
   10 CONTINUE                                                          05470002
      GO TO 35                                                          05480002
   25 KCIP(J)=1                                                         05490002
      GO TO 40                                                          05500002
   35 KCIP(J)=0                                                         05510002
   40 RETURN                                                            05520002
      END                                                               05530002
C *** THIS SUBROUTINE IS CALLED WHEN A NEW FSCM IS READ.                05540002
C     THE CONTENTS OF THE NEW FSCM ARE MOVED TO THE FIRST POSITION OF   05550002
C     FACILITY ARRAY                                                    05560002
C*********************************************************            05570002
      SUBROUTINE NEWFAC(J,DCASR,QAORG,FSCM,TYP,CMDTY,PVN,MONTH,YEAR,    05580002
     1PLANHR,LOTINS,PEHR,AQDR,BQDR,CQDR,DQDR,EQDR,ADMNHR,SHIPMT,WD,INTHRO5590002
     2,REINHR,VISIT,ECP,MTGHR,MRB,PCO,CAO,SI,EPA,ALRR,CONTR,DLRIN,DLROUTO5600002
     3,ACNTRT,BCNTRT,OCNTRT,QALIIN,QALIRE,AONHND,BONHND,OONHND,DLROH,    05610002
     4NQDR,PVINP,PELONP,PEELNP,NQAR,OPER,WMDR,DAYSCL,ISEQ,RECS,ISTRAT,  05620002
     5JCIP,ICIPNO,ICIP,PVIHR)                                           05630002
C                                                                       05640002
      CHARACTER DCASR(120)*6,QAORG(120)*3,FSCM(120)*6,TYP(120)*1,       05650002
     1CMDTY(120)*2,PVN(120)*1,OPER(120)*2,BFSCM(2000)*5                 05660002
      INTEGER MONTH(120),YEAR(120),PLANHR(120),LOTINS(120),AQDR(120),   05670002
     1BQDR(120),CQDR(120),DQDR(120),EQDR(120),ADMNHR(120),SHIPMT(120),  05680002
     2WD(120),INTHR(120),REINHR(120),VISIT(120),ECP(120),MTGHR(120),    05690002
     3MRB(120),PCO(120),CAO(120),CONTR(120),JCIP(120),ISTRAT(120),      05700002
     4DLRIN(120),DLROUT(120),ACNTRT(120),BCNTRT(120),OCNTRT(120),       05710002
     5QALIIN(120),QALIRE(120),AONHND(120),BONHND(120),OONHND(120),      05720002
     6DLROH(120),NQDR(120),PVINP(120),PELONP(120),PEELNP(120),NQAR(120),05730002
     7ISEQ(120),RECS(120),DAYSCL(120),PEHR(120),PVIHR(120)              05740002
      REAL WMDR(120),SI(120),EPA(120),ALRR(120)                         05750002
      COMMON /CHTR/BFSCM,FLAG                                           05760002
      IF (ICIP .EQ. 1) GOTO 1                                           05770002
      CALL CIP(J,FSCM,JCIP,ICIPNO)                                      05780002
    1 DCASR(1)=DCASR(J)                                                 05790002
      QAORG(1)=QAORG(J)                                                 05800002
      FSCM(1)=FSCM(J)                                                   05810002
      TYP(1)=TYP(J)                                                     05820002
      PVN(1)=PVN(J)                                                     05830002
      CMDTY(1)=CMDTY(J)                                                 05840002
      MONTH(1)=MONTH(J)                                                 05850002
      YEAR(1)=YEAR(J)                                                   05860002
      PLANHR(1)=PLANHR(J)                                               05870002
      PVIHR(1)=PVIHR(J)                                                 05880002
      LOTINS(1)=LOTINS(J)                                               05890002
      PEHR(1)=PEHR(J)                                                   05900002
      AQDR(1)=AQDR(J)                                                   05910002
```

```
      BQDR(1)=BQDR(J)                                                   05920002
      CQDR(1)=CQDR(J)                                                   05930002
      DQDR(1)=DQDR(J)                                                   05940002
      EQDR(1)=EQDR(J)                                                   05950002
      ADMNHR(1)=ADMNHR(J)                                               05960002
      SHIPMT(1)=SHIPMT(J)                                               05970002
      WD(1)=WD(J)                                                       05980002
      INTHR(1)=INTHR(J)                                                 05990002
      REINHR(1)=REINHR(J)                                               06000002
      VISIT(1)=VISIT(J)                                                 06010002
      ECP(1)=ECP(J)                                                     06020002
      MTGHR(1)=MTGHR(J)                                                 06030002
      MRB(1)=MRB(J)                                                     06040002
      PCO(1)=PCO(J)                                                     06050002
      CAO(1)=CAO(J)                                                     06060002
      SI(1)=SI(J)                                                       06070002
      EPA(1)=EPA(J)                                                     06080002
      ALRR(1)=ALRR(J)                                                   06090002
      CONTR(1)=CONTR(J)                                                 06100002
      DLRIN(1)=DLRIN(J)                                                 06110002
      DLROUT(1)=DLROUT(J)                                               06120002
      ACNTRT(1)=ACNTRT(J)                                               06130002
      BCNTRT(1)=BCNTRT(J)                                               06140002
      OCNTRT(1)=OCNTRT(J)                                               06150002
      QALIIN(1)=QALIIN(J)                                               06160002
      QALIRE(1)=QALIRE(J)                                               06170002
      AONHND(1)=AONHND(J)                                               06180002
      BONHND(1)=BONHND(J)                                               06190002
      OONHND(1)=OONHND(J)                                               06200002
      DLROH(1)=DLROH(J)                                                 06210002
      NQDR(1)=NQDR(J)                                                   06220002
      PVINP(1)=PVINP(J)                                                 06230002
      PELONP(1)=PELONP(J)                                               06240002
      PEELNP(1)=PEELNP(J)                                               06250002
      NQAR(1)=NQAR(J)                                                   06260002
      OPER(1)=OPER(J)                                                   06270002
      ISTRAT(1)=ISTRAT(J)                                               06280002
      WMDR(1)=WMDR(J)                                                   06290002
      DAYSCL(1)=DAYSCL(J)                                               06300002
      ISEQ(1)=ISEQ(J)                                                   06310002
      RECS(1)=RECS(J)                                                   06320002
      JCIP(1)=JCIP(J)                                                   06330002
      RETURN                                                            06340002
      END                                                               06350002
C     ****************************************************************  06360002
C     THIS SUBROUTINE COMPUTES ATTRIBUTE VALUES AND RATES PRIOR TO      06370002
C     TOPSIS PROCESSING.                                                06380002
C                                                                       06390002
      SUBROUTINE PREPIN(KA,AQDR,BQDR,CQDR,DQDR,EQDR,WD,ECP,MRB,EPA,ALRR,06400002
     1WMDR,TOPCA,CARATE,WDRATE,ECPRAT,AMRBRA,EPARAT,ALRRRA,WMDRRA,      06410002
     2STARTM,STARTY,ENDMO,ENDYR,FSCM,MONTH,YEAR,TOPWD,TOPECP,TOPMRB,    06420002
     3TOPEPA,TOPLRR,TOPMDR,DCASR,QADRG,DEGREE,AIDEAL,AWORST,BIDEAL,     06430002
     4BWORST,ISTRAT,WEIGHT,REINHR,INTHR,MTGHR,SHIPMT,LOTINS,VISIT,PVN,  06440002
     5TYP,PEHR,CONTR,PLANHR,DAYSCL,PCO,CAO,NQDR,SI,ADMNHR,PVINP,PELONP, 06450002
     6PEELNP,QALIIN,QALIRE,LONG,TOPSCR,PVIHR)                           06460002
      INTEGER AQDR(120),BQDR(120),CQDR(120),DQDR(120),EQDR(120),WD(120),06470002
     1ECP(120),MRB(120),STARTM,STARTY,ENDMO,ENDYR,MONTH(120),YEAR(120), 06480002
     2IMONTH(120),IYEAR(120),DEGREE(120),IDGREE(120),ISTRAT(120),       06490002
     3JSTRAT(120),REINHR(120),INTHR(120),MTGHR(120),SHIPMT(120),        06500002
     4LOTINS(120),VISIT(120),PEHR(120),CONTR(120),PLANHR(120),          06510002
     5DAYSCL(120),PCO(120),CAO(120),NQDR(120),ADMNHR(120),PVINP(120),   06520002
     6PELONP(120),PEELNP(120),QALIIN(120),QALIRE(120),FLAGA(120)        06530002
      INTEGER FLAGB(120),FLAGC1(120),FLAGC2(120),FLAGC3(120),FLAGC4(120)06540002
     1,FLAGC5(120),FLAGD(120),FLAGEF(120),FLAGE(120),FLAGF(120),        06550002
     2FLAGG2(120),FLAGH1(120),FLAGH2(120),FLAGJ(120),FLAGK1(120)        06560002
      INTEGER FLAGK2(120),FLAGL(120),FLAGN(120),FLAGO(120),FLAGP(120),  06570002
```

```
     1FLAGO1(120),FLAGO2(120),FLAGO3(120),FCOUNT(120),FLAGE1(120),     06580002
                                                                       06590002
     2PVIHR(120)                                                       06600002
      CHARACTER FSCM(120)*6,DCASR(120)*6,OAORG(120)*3,DCASCD(120)*6,   06610002
     1ORGCD(120)*3,PVN(120)*1,TYP(120)*1,FLAGF1(120)*1,FLAGG1(120)*1,  06620002
     2FLAG(120)*18,TP(120)*1                                           06630002
      REAL WMOR(120),SI(120),EPA(120),ALRR(120),TOPCA(120),CARATE(120), 06640002
     1WDRATE(120),ECPRAT(120),AMRBRA(120),EPARAT(120),ALRRRA(120),     06650002
     2WMDRRA(120),TOPEPA(120),TOPLRR(120),TOPMRB(120),TOPECP(120),     06660002
     3TOPWD(120),TOPMDR(120),A(628,36),TOPSCR(12,120),AIDEAL(4,7),     06670002
     4AWORST(4,7),BIDEAL(4,7),BWORST(4,7),WEIGHT(4,7),FLAGM(120)       06680002
      COMMON /RVAR/A                                                   06690002
      COMMON /CHTR/BFSCM,FLAG                                          06700014
C    INITIALIZE ARRAYS. ATTRIBUTE AND RATE DEFAULT VALUES ARE 0.0.     06710002
      DO 5 I=1,120                                                     06720002
       TOPCA(I)=0.0                                                    06730002
       CARATE(I)=0.0                                                   06740002
       TOPEPA(I)=0.0                                                   06750002
       EPARAT(I)=0.0                                                   06760002
       TOPLRR(I)=0.0                                                   06770002
       ALRRRA(I)=0.0                                                   06780002
       TOPMRB(I)=0.0                                                   06790002
       AMRBRA(I)=0.0                                                   06800002
       TOPECP(I)=0.0                                                   06810002
       ECPRAT(I)=0.0                                                   06820002
       TOPWD(I)=0.0                                                    06830002
       WDRATE(I)=0.0                                                   06840002
       TOPMDR(I)=0.0                                                   06850002
       WMDRRA(I)=0.0                                                   06860002
     5 CONTINUE                                                        06870002
C    COMPUTE THE NUMBER OF MONTHS TOPSIS WILL PROCESS.                 06880002
        MON=(ENDYR-STARTY)*12 + ENDMO - STARTM + 1                     06890002
        IF((YEAR(KA) .LT. ENDYR) .OR. ((YEAR(KA) .EQ. ENDYR) .AND.     06900002
      1 (MONTH(KA) .LT. ENDMO))) MON=MON-((ENDYR-YEAR(KA))*12+ENDMO-   06910002
      2MONTH(KA))                                                      06920002
        IF (MON .GE. KA) GOTO 30                                       06930002
        DO 10 I=1,MON                                                  06940002
C    **** COMPUTE RATES FOR ATTRIBUTES.RATES RANGE FROM -3 TO +3.      06950002
C    RATE IS COMPUTED BY TAKING RATIO OF FIRST HALF TO SECOND HALF     06960002
C    OF DATA ARRAY. RATES OF 0 TO 3 ARE INCREASES.                     06970002
C    INITIALIZE INTERNAL PARAMETERS                                    06980002
         IPRICA=0                                                      06990002
         IAFTCA=0                                                      07000002
         PRIEPA=0.0                                                    07010002
         AFTEPA=0.0                                                    07020002
         IPRIMR=0                                                      07030002
         IAFTMR=0                                                      07040002
         IPRIWD=0                                                      07050002
         IAFTWD=0                                                      07060002
         PRILRR=0.0                                                    07070002
         AFTLRR=0.0                                                    07080002
         IPRIEC=0                                                      07090002
         IAFTEC=0                                                      07100002
         PRIMDR=0.0                                                    07110002
         AFTMDR=0.0                                                    07120002
C     IF THERE IS MORE THAN 12 MONTHS OF DATA, MODEL IGNORES PRIOR     07130002
C      DATA WHEN COMPUTING RATES.                                      07140002
         JMON=MON-I                                                    07150002
         KC=KA-JMON                                                    07160002
         IF(KC .GT. 12) GOTO 7                                         07170002
         IOFFST=1                                                      07180002
         GO TO 6                                                       07190002
     7   IOFFST=KC-11                                                  07200002
         KC=12                                                         07210002
     6   MIDMON=(KC + 1)/2                                             07220002
         IF (MIDMON*2 .EQ. (KC +1)) GOTO 22                            07230002
C      EVEN NUMBER OF DATA POINTS
```

```
                                                                        07240002
      INDMO=MIDMON+IOFFST-1                                             07250002
      GO TO 21                                                          07260002
C     ODD NUMBER OF DATA POINTS                                         07270002
   22 INDMO=MIDMON+IOFFST-2                                             07280002
   21 DO 24 J=IOFFST,INDMO                                              07290002
      K=MIDMON + J                                                      07300002
      IPRICA=IPRICA+(AQDR(J)*3+BQDR(J)*10+CQDR(J)*30+DQDR(J)*60         07310002
     1+EQDR(J)*30)                                                      07320002
      IAFTCA=IAFTCA+(AQDR(K)*3+BQDR(K)*10+CQDR(K)*30+DQDR(K)*60         07330002
     1+EQDR(K)*30)                                                      07340002
      PRIEPA=PRIEPA+EPA(J)                                              07350002
      AFTEPA=AFTEPA+EPA(K)                                              07360002
      IPRIMR=IPRIMR+MRB(J)                                              07370002
      IAFTMR=IAFTMR+MRB(K)                                              07380002
      IPRIWD=IPRIWD+WD(J)                                               07390002
      IAFTWD=IAFTWD+WD(K)                                               07400002
      PRILRR=PRILRR+ALRR(J)                                             07410002
      AFTLRR=AFTLRR+ALRR(K)                                             07420002
      IPRIEC=IPRIEC+ECP(J)                                              07430002
      IAFTEC=IAFTEC+ECP(K)                                              07440002
      PRIMDR=PRIMDR+WMDR(J)                                             07450002
      AFTMDR=AFTMDR+WMDR(K)                                             07460002
   24 CONTINUE                                                          07470002
      I=KA-MON+I                                                        07480002
C     **** COMPUTE ATTRIBUTE VALUES.                                    07490002
      TOPCA(I)=AQDR(L)*3+BQDR(L)*10+CQDR(L)*30+DQDR(L)*60+EQDR(L)*30    07500002
      TOPEPA(I)=EPA(L)                                                  07510002
      TOPMRB(I)=MRB(L)                                                  07520002
      TOPWD(I)=WD(L)                                                    07530002
      TOPLRR(I)=ALRR(L)                                                 07540002
      TOPECP(I)=ECP(L)                                                  07550002
      TOPMDR(I)=WMDR(L)                                                 07560002
C  IF SECOND HALF IS NEGLIGIBLE OR ZERO, RATE IS -3 OR 0 DEPENDING ON   07570002
C  FIRST HALF ACTIVITY.                                                 07580002
C  RATE IS A VALUE BETWEEN -3 AND +3                                    07590002
   26 CARATE(I)=(REAL(IPRICA))/(REAL(IAFTCA)+.1)*3.0                    07600002
      IF(CARATE(I) .GT. 6.0) CARATE(I)=6.0                              07610002
      IF((IPRICA+IAFTCA) .EQ. 0) CARATE(I)=3.0                          07620002
      CARATE(I)=3.0-CARATE(I)                                           07630002
      EPARAT(I)=PRIEPA/(AFTEPA+.01)*3.0                                 07640002
      IF(EPARAT(I) .GT. 6.0) EPARAT(I)=6.0                              07650002
      IF((PRIEPA+AFTEPA) .EQ. 0.0) EPARAT(I)=3.0                        07660002
      EPARAT(I)=3.0-EPARAT(I)                                           07670002
      AMRBRA(I)=(REAL(IPRIMR))/(REAL(IAFTMR)+.01)*3.0                   07680002
      IF(AMRBRA(I) .GT. 6.0) AMRBRA(I)=6.0                              07690002
      IF((IPRIMR+IAFTMR) .EQ. 0) AMRBRA(I)=3.0                          07700002
      AMRBRA(I)=3.0-AMRBRA(I)                                           07710002
      WDRATE(I)=(REAL(IPRIWD))/(REAL(IAFTWD)+.01)*3.0                   07720002
      IF(WDRATE(I) .GT. 6.0) WDRATE(I)=6.0                              07730002
      IF((IPRIWD+IAFTWD) .EQ. 0) WDRATE(I)=3.0                          07740002
      WDRATE(I)=3.0-WDRATE(I)                                           07750002
      ALRRRA(I)=PRILRR/(AFTLRR+.01)*3.0                                 07760002
      IF(ALRRRA(I) .GT. 6.0) ALRRRA(I)=6.0                              07770002
      IF((PRILRR+AFTLRR) .EQ. 0.0) ALRRRA(I)=3.0                        07780002
      ALRRRA(I)=3.0-ALRRRA(I)                                           07790002
      ECPRAT(I)=(REAL(IPRIEC))/(REAL(IAFTEC)+.01)*3.0                   07800002
      IF(ECPRAT(I) .GT. 6.0) ECPRAT(I)=6.0                              07810002
      IF((IPRIEC+IAFTEC) .EQ. 0) ECPRAT(I)=3.0                          07820002
      ECPRAT(I)=3.0-ECPRAT(I)                                           07830002
      WMDRRA(I)=PRIMDR/(AFTMDR+.01)*3.0                                 07840002
      IF(WMDRRA(I) .GT. 6.0) WMDRRA(I)=6.0                              07850002
      IF((PRIMDR+AFTMDR) .EQ. 0.0) WMDRRA(I)=3.0                        07860002
      WMDRRA(I)=3.0-WMDRRA(I).                                          07870002
C SHIFT OTHER VARIABLES TO THE NEW TIME FRAME                           07880002
      CALL SHIFTR(I,L,DCASCD,DCASR,ORGCD,OAORG,IMONTH,MONTH,IYEAR,YEAR, 07890002
     1IDGREE,DEGREE,JSTRAT,ISTRAT,FLAGA,REINHR,FLAGB,INTHR,FLAGD,MTGHR,
```

```
     2FLAGEF,SHIPMT,FLAGE,LOTINS,FLAGF,VISIT,FLAGF1,TYP,TP,FLAGG1,PVN,   07900002
     3FLAGG2,PEHR,FLAGH1,CONTR,FLAGH2,PLANHR,FLAGJ,DAYSCL,FLAGK1,QALIIN,07910002
     4FLAGK2,QALIRE,FLAGL,NQOR,FLAGM,SI,FLAGN,PCO,FLAGO,CAO,FLAGP,       07920002
     5ADMNHR,FLAGQ1,PVINP,FLAGQ2,PELONP,FLAGQ3,PEELNP,FLAGC1,AQOR,       07930002
     6FLAGC2,BQOR,FLAGC3,CQOR,FLAGC4,DQOR,FLAGC5,EQOR,FLAGE1,PVIHR)      07940002
                                                                         07950002
C  COMPUTE TOPSIS SCORES                                                07960002
      CALL TOPSIS(I,IDGREE,TOPCA,CARATE,TOPEPA,EPARAT,TOPMRB,AMRBRA,     07970002
     1TOPWD,WDRATE,TOPLRR,ALRRRA,TOPECP,ECPRAT,TOPMDR,WMDRRA,TOPSCR,     07980002
     2AIDEAL,AWORST,BIDEAL,BWORST,JSTRAT,WEIGHT)                         07990002
C  IDENTIFY RED FLAG CONDITIONS                                         08000002
      CALL FLGGR(I,FLAGA,TOPCA,FLAGB,FLAGC1,FLAGC2,FLAGC3,FLAGC4,        08010002
     1FLAGC5,FLAGD,FLAGEF,FLAGE,FLAGE1,                                  08020002
     2FLAGF,FLAGF1,FLAGG1,FLAGG2,FLAGH1,                                 08030002
     3FLAGH2,TOPLRR,FLAGJ,FLAGK1,FLAGK2,FLAGL,FLAGM,FLAGN,FLAGO,FLAGP,   08040002
     4FLAGQ1,FLAGQ2,FLAGQ3,FCOUNT,JSTRAT,TOPEPA)                         08050002
      CALL SCORER(I,FCOUNT,TOPSCR,TP,JSTRAT)                             08060002
C  WRITE RECORD TO VERIFY PROGRAM                                       08070002
       IF (LONG .EQ. 0) GOTO 50                                          08080002
        WRITE(9,25) I,DCASCD(I),ORGCD(I),FSCM(1),IMONTH(I),IYEAR(I),     08090002
     1JSTRAT(I),IDGREE(I),TOPEPA(I),EPARAT(I),TOPLRR(I),ALRRRA(I),       08100002
     2TOPMRB(I),AMRBRA(I),TOPWD(I),WDRATE(I),TOPECP(I),ECPRAT(I),        08110002
     3TOPCA(I),CARATE(I),TOPMDR(I),WMDRRA(I),                            08120002
     4TOPSCR(1,I),TOPSCR(2,I),TOPSCR(3,I),TOPSCR(4,I),TOPSCR(5,I),       08130002
     5TOPSCR(6,I),TOPSCR(7,I),TOPSCR(8,I),TOPSCR(9,I),TOPSCR(10,I),      08140002
     6TOPSCR(11,I),FLAG(I),FCOUNT(I)                                     08150002
   25  FORMAT(I3,A6,A3,A6,2I3,I4,I2,2(F6.2,F5.2),5(F4.0,F5.2),11F5.1,     08160002
     1A18,I2)                                                            08170002
   50  IF((IYEAR(I) .LT. STARTY) .OR. ((IYEAR(I) .EQ. STARTY) .AND.       08180002
     1(IMONTH(I) .LT. STARTM))) GOTO 10                                  08190002
       WRITE(11,51) ORGCD(I),FSCM(1),TP(I),IMONTH(I),IYEAR(I),           08200002
     1JSTRAT(I),IDGREE(I),FLAG(I),TOPSCR(10,I),                          08210002
     2TOPSCR(1,I),TOPSCR(2,I),TOPSCR(3,I),TOPSCR(4,I),TOPSCR(5,I),       08220002
     3TOPSCR(6,I),TOPSCR(7,I),TOPSCR(9,I),TOPSCR(11,I),TOPSCR(12,I)      08230002
   51  FORMAT(A3,A6,A1,2I3,I4,I2,A18,7F7.1,F12.1,3F7.1)                   08240002
   10  CONTINUE                                                          08250002
       GO TO 40                                                          08260002
   30 WRITE (6,35) FSCM(1),KA,MON                                        08270002
   35 FORMAT(2X,'WARNING. INSUFFICIENT DATA FOR FSCM',A6,2I3,            08280002
     1'FSCM SKIPPED BUT DATA ON THE LABEL FILE')                         08290002
   40 RETURN                                                             08300002
      END                                                                08310002
      SUBROUTINE TOPSIS(I,IDGREE,TOPCA,CARATE,TOPEPA,EPARAT,TOPMRB,      08320002
     1AMRBRA,TOPWD,WDRATE,TOPLRR,ALRRRA,TOPECP,ECPRAT,TOPMDR,WMDRRA,     08330002
     2TOPSCR,AIDEAL,AWORST,BIDEAL,BWORST,JSTRAT,WEIGHT)                  08340002
      REAL A(628,36),TOPCA(120),CARATE(120),TOPEPA(120),EPARAT(120),     08350002
     1TOPMRB(120),AMRBRA(120),TOPWD(120),WDRATE(120),TOPLRR(120),        08360002
     2ALRRRA(120),TOPECP(120),ECPRAT(120),TOPMDR(120),WMDRRA(120),       08370002
     3TOPSCR(12,120),AIDEAL(4,7),AWORST(4,7),BIDEAL(4,7),BWORST(4,7),    08380002
     4Z(7),SPLUS(7),SMINUS(7),WEIGHT(4,7)                                08390002
      INTEGER IDGREE(120),JSTRAT(120)                                    08400002
      COMMON /RVAR/A                                                     08410002
      TOPSCR(8,I)=0.0                                                    08420002
      WTSUM=0.0                                                          08430002
      SPOS=0.0                                                           08440002
      SNEG=0.0                                                           08450002
C COMPUTE Z VALUES FOR NONRATE PARAMETERS                                08460002
C  NEXT LINES ASSUME EXPONENTIAL DISTRIBUTION                            08470002
      Z(1)=LOG(TOPEPA(I)/(A(JSTRAT(I),13)+.0001)+.00001)                 08480002
      Z(2)=LOG(TOPLRR(I)/(A(JSTRAT(I),15)+.0001)+.00001)                 08490002
      Z(3)=LOG(TOPMRB(I)/(A(JSTRAT(I),7)+.0001)+.00001)                  08500002
      Z(4)=LOG(TOPWD(I)/(A(JSTRAT(I),3)+.0001)+.00001)                   08510002
      Z(5)=LOG(TOPECP(I)/(A(JSTRAT(I),5)+.0001)+.00001)                  08520002
      Z(6)=LOG(TOPCA(I)/(A(JSTRAT(I),23)+.0001)+.00001)                  08530002
      Z(7)=LOG(TOPMDR(I)/(A(JSTRAT(I),33)+.0001)+.00001)                 08540002
C ASIGN LIMITS TO Z VALUES FOR OUTLIERS. Z MUST BE BETWEEN -3.0 AND 3.0  08550002
      DO 1 J=1,7
```

```
            IF(Z(J) .LT. -3.0) Z(J)=-3.0                                    08560002
            IF(Z(J) .GT. 3.0) Z(J)=3.0                                      08570002
                                                                            08580002
          1 CONTINUE                                                        08590002
C COMPUTE TOPSIS SEPARATION MEASURES FROM NEGATIVE IDEAL                    08600002
            SMINUS(1)=BWORST(IDGREE(I),1)-EPARAT(I)                         08610002
            SMINUS(2)=BWORST(IDGREE(I),2)-ALRRRA(I)                         08620002
            SMINUS(3)=BWORST(IDGREE(I),3)-AMRBRA(I)                         08630002
            SMINUS(4)=BWORST(IDGREE(I),4)-WDRATE(I)                         08640002
            SMINUS(5)=BWORST(IDGREE(I),5)-ECPRAT(I)                         08650002
            SMINUS(6)=BWORST(IDGREE(I),6)-CARATE(I)                         08660002
            SMINUS(7)=BWORST(IDGREE(I),7)-WMDRRA(I)                         08670002
           IF((IDGREE(I) .EQ. 1) .OR. (IDGREE(I) .EQ. 3)) THEN              08680002
             SMINUS(6)=Z(6)-AWORST(IDGREE(I),6)                             08690002
             SMINUS(7)=AWORST(IDGREE(I),7)-Z(7)                             08700002
           ELSE                                                             08710002
            DO 2 J=1,7                                                      08720014
C    BELOW LINE REFLECTS 'CITY BLOCK' DISTANCE.                             08730002
C    CITY BLOCK SEEMS TO WORK BETTER FOR NONRESIDENT.                       08740002
              SMINUS(J)=SMINUS(J)+(AWORST(IDGREE(I),J)-Z(J))                08750002
          2   CONTINUE                                                      08760002
           ENDIF                                                            08770002
C COMPUTE TOPSIS SEPARATION MEASURES FROM POSITIVE IDEAL                    08780002
            SPLUS(1)=EPARAT(I)-BIDEAL(IDGREE(I),1)                          08790002
            SPLUS(2)=ALRRRA(I)-BIDEAL(IDGREE(I),2)                          08800002
            SPLUS(3)=AMRBRA(I)-BIDEAL(IDGREE(I),3)                          08810002
            SPLUS(4)=WDRATE(I)-BIDEAL(IDGREE(I),4)                          08820002
            SPLUS(5)=ECPRAT(I)-BIDEAL(IDGREE(I),5)                          08830002
            SPLUS(6)=CARATE(I)-BIDEAL(IDGREE(I),6)                          08840002
            SPLUS(7)=WMDRRA(I)-BIDEAL(IDGREE(I),7)                          08850002
           IF((IDGREE(I) .EQ. 1) .OR. (IDGREE(I) .EQ. 3)) THEN              08860002
             SPLUS(6)=AIDEAL(IDGREE(I),6)-Z(6)                              08870002
             SPLUS(7)=Z(7)-AIDEAL(IDGREE(I),7)                              08880002
           ELSE                                                             08890002
            DO 7 J=1,7                                                      08900014
C    BELOW LINE REFLECTS 'CITY BLOCK' DISTANCE.                             08910002
C    CITY BLOCK SEEMS TO WORK BETTER FOR NONRESIDENT.                       08920002
              SPLUS(J)=SPLUS(J)+(Z(J)-AIDEAL(IDGREE(I),J))                  08930002
          7   CONTINUE                                                      08940002
           ENDIF                                                            08950002
C  COMBINE DISTANCES INTO AN OVERALL TOPSIS SCORE                           08960002
            DO 9 J=1,7                                                      08970002
            WTSUM=WTSUM+WEIGHT(IDGREE(I),J)                                 08980002
            SNEG=SNEG+(SMINUS(J)*WEIGHT(IDGREE(I),J))**2                    08990002
            SPOS=SPOS+(SPLUS(J)*WEIGHT(IDGREE(I),J))**2                     09000002
            TOPSCR(J,I)=SMINUS(J)/(SPLUS(J)+SMINUS(J)+.00001)*100           09010002
            TOPSCR(8,I)= TOPSCR(8,I)+TOPSCR(J,I)*WEIGHT(IDGREE(I),J)        09020002
          9 CONTINUE                                                        09030002
            TOPSCR(9,I)=SQRT(SNEG)/(SQRT(SNEG)+SQRT(SPOS))*100              09040002
            TOPSCR(8,I)=TOPSCR(8,I)/WTSUM                                   09050002
C  COMPUTE A MODIFIED SCORE BASED ON DEGREE OF DIFFICULTY                   09060002
            CALL LIMITR(I,IDGREE,TOPSCR)                                    09070002
            RETURN                                                          09080002
            END                                                             09090002
C                                                                           09100002
C  *** DEFINES IDEAL AND NEGATIVE IDEAL CONDITIONS FOR EACH DIFFICULTY      09110002
C       ALSO DETERMINES WEIGHT FACTORS FOR EACH SITUATION                   09120002
C                                                                           09130002
            SUBROUTINE CORNER(AIDEAL,AWORST,BIDEAL,BWORST,WEIGHT)           09140002
            REAL AIDEAL(4,7),AWORST(4,7),BIDEAL(4,7),BWORST(4,7),WEIGHT(4,7) 09150002
C DEFINE IDEAL AND NEGATIVE IDEALS                                          09160002
C J=1 - EPA,J=2 - LRR,J=3 - MRB,J=4 - WD,J=5 - ECP,J=6 - CA,J=7 -MDR        09170002
C K IS DIFFICULTY INDEX                                                     09180002
            DO 1 K=1,4                                                      09190002
             DO 2 J=1,7                                                     09200002
              AIDEAL(K,J)=-3.0                                              09210002
              AWORST(K,J)=3.0
```

```
      BIDEAL(K,J)=-3.0                                                  09220002
      BWORST(K,J)=3.0                                                   09230002
    2 CONTINUE                                                          09240002
      IF((K .EQ. 1) .OR. (K .EQ. 3)) THEN                               09250002
       DO 3 J=1,5                                                       09260002
        AIDEAL(K,J)=0.0                                                 09270002
        AWORST(K,J)=0.0                                                 09280002
    3  CONTINUE                                                         09290002
       AIDEAL(K,6)=3.0                                                  09300002
       AWORST(K,6)=-3.0                                                 09310002
       BIDEAL(K,6)=0.0                                                  09320002
       BWORST(K,6)=0.0                                                  09330002
       BIDEAL(K,7)=0.0                                                  09340002
       BWORST(K,7)=0.0                                                  09350002
      ENDIF                                                             09360002
      WEIGHT(K,1)=.77                                                   09370002
      WEIGHT(K,2)=.77                                                   09380002
      WEIGHT(K,3)=.91                                                   09390002
      WEIGHT(K,4)=.72                                                   09400002
      WEIGHT(K,5)=.32                                                   09410002
      WEIGHT(K,6)=1.00                                                  09420002
      WEIGHT(K,7)=.98                                                   09430002
    1 CONTINUE                                                          09440002
      DO 9 K=3,4                                                        09450002
       WEIGHT(K,1)=.66                                                  09460002
       WEIGHT(K,2)=.98                                                  09470002
       WEIGHT(K,3)=.83                                                  09480002
       WEIGHT(K,4)=.64                                                  09490002
    9 CONTINUE                                                          09500002
      RETURN                                                            09510002
      END                                                               09520002
C                                                                       09530002
C THIS SUBROUTINE ESTABLISHES ARBITRARY LIMITS ON TOPSIS SCORES         09540002
C                                                                       09550002
      SUBROUTINE LIMITR(I,IDGREE,TOPSCR)                                09560002
      REAL TOPSCR(12,120)                                               09570002
      INTEGER IDGREE(120)                                               09580002
      COMMON /RVAR/A                                                    09590002
      DO 2 J=8,9                                                        09600002
       IF ((IDGREE(I) .EQ. 1) .OR. (IDGREE(I) .EQ. 3)) TOPSCR(J,I)=     09610002
     1TOPSCR(J,I)*.79                                                   09620002
    2 CONTINUE                                                          09630002
      RETURN                                                            09640002
      END                                                               09650002
C                                                                       09660002
C THIS SUBROUTINE LINKS HISTORY ARRAYS WITH MODEL ARRAYS                09670002
C                                                                       09680002
      SUBROUTINE SHIFTR(I,L,DCASCD,DCASR,ORGCD,QADRG,IMNTH,MNTH,IYR,YR, 09690002
     1IDGREE,DEGREE,JSTRAT,ISTRAT,FLAGA,REINHR,FLAGB,INTHR,FLAGD,MTGHR, 09700002
     2FLAGEF,SHIPMT,FLAGE,LOTINS,FLAGF,VISIT,FLAGF1,TYP,TP,FLAGG1,PVN,  09710002
     3FLAGG2,PEHR,FLAGH1,CONTR,FLAGH2,PLANHR,FLAGJ,DAYSCL,FLAGK1,QALIIN,09720002
     4FLAGK2,QALIRE,FLAGL,NQDR,FLAGM,SI,FLAGN,PCO,FLAGO,CAO,FLAGP,      09730002
     5ADMNHR,FLAGQ1,PVINP,FLAGQ2,PELONP,FLAGQ3,PEELNP,FLAGC1,AQDR,      09740002
     6FLAGC2,BQDR,FLAGC3,CQDR,FLAGC4,DQDR,FLAGC5,EQDR,FLAGE1,PVIHR)     09750002
      INTEGER MNTH(120),YR(120),AQDR(120),BQDR(120),CQDR(120),DQDR(120),09760002
     1IMNTH(120),IYR(120),DEGREE(120),IDGREE(120),ISTRAT(120),          09770002
     2JSTRAT(120),REINHR(120),INTHR(120),MTGHR(120),SHIPMT(120),        09780002
     3LOTINS(120),VISIT(120),PEHR(120),CONTR(120),PLANHR(120),          09790002
     4DAYSCL(120),PCO(120),CAO(120),NQDR(120),ADMNHR(120),PVINP(120),   09800002
     5PELONP(120),PEELNP(120),QALIIN(120),QALIRE(120),FLAGA(120),       09810002
     6FLAGB(120),FLAGD(120),FLAGEF(120),FLAGE(120),FLAGF(120),          09820002
     7FLAGG2(120),FLAGH1(120),FLAGH2(120),FLAGJ(120),FLAGK1(120),       09830002
     8FLAGK2(120),FLAGL(120),FLAGM(120),FLAGN(120),FLAGO(120),          09840002
     9FLAGQ1(120),FLAGQ2(120),FLAGQ3(120),EQDR(12),FLAGC1(120)          09850002
      INTEGER FLAGC2(120),FLAGC3(120),FLAGC4(120),FLAGC5(120),          09860002
     1FLAGE1(120),PVIHR(120)                                            09870002
```

```
      CHARACTER DCASR(120)*6,QAORG(120)*3,DCASCD(120)*6,TP(120)*1,      09880002
     1ORGCD(120)*3,PVN(120)*1,TYP(120)*1,FLAGF1(120)*1,FLAGG1(120)*1    09890002
      REAL SI(120),FLAGM(120)                                           09900002
        DCASCD(I)=DCASR(L)                                              09910002
        ORGCD(I)=QAORG(L)                                               09920002
        IMNTH(I)=MNTH(L)                                                09930002
        IYR(I)=YR(L)                                                    09940002
        IDGREE(I)=DEGREE(L)                                             09950002
        JSTRAT(I)=ISTRAT(L)                                             09960002
        TP(I)=TYP(L)                                                    09970002
        FLAGA(I)=REINHR(L)                                              09980002
        FLAGB(I)=INTHR(L)                                               09990002
        FLAGC1(I)=AQDR(L)                                               10000002
        FLAGC2(I)=BQDR(L)                                               10010002
        FLAGC3(I)=CQDR(L)                                               10020002
        FLAGC4(I)=DQDR(L)                                               10030002
        FLAGC5(I)=EQDR(L)                                               10040002
        FLAGD(I)=MTGHR(L)                                               10050002
        FLAGEF(I)=SHIPMT(L)                                             10060002
        FLAGE(I)=LOTINS(L)                                              10070002
        FLAGE1(I)=PVIHR(L)                                              10080002
        FLAGF(I)=VISIT(L)                                               10090002
        FLAGF1(I)=TYP(L)                                                10100002
        FLAGG1(I)=PVN(L)                                                10110002
        FLAGG2(I)=PEHR(L)                                               10120002
        FLAGH1(I)=CONTR(L)                                              10130002
        FLAGH2(I)=PLANHR(L)                                             10140002
        FLAGJ(I)=DAYSCL(L)                                              10150002
        FLAGK1(I)=QALIIN(L)                                             10160002
        FLAGK2(I)=QALIRE(L)                                             10170002
        FLAGL(I)=NQDR(L)                                                10180002
        FLAGM(I)=SI(L)                                                  10190002
        FLAGN(I)=PCO(L)                                                 10200002
        FLAGO(I)=CAO(L)                                                 10210002
        FLAGP(I)=ADMNHR(L)                                              10220002
        FLAGQ1(I)=PVINP(L)                                              10230002
        FLAGQ2(I)=PELONP(L)                                             10240002
        FLAGQ3(I)=PEELNP(L)                                             10250002
        RETURN                                                          10260002
        END                                                             10270002
C   *** THIS SUBROUTINE CHECKS FOR OUT OF TOLERANCE CONDITIONS***       10280002
        SUBROUTINE FLGGR(I,FLAGA,TOPCA,FLAGB,FLAGC1,FLAGC2,FLAGC3,      10290002
     1FLAGC4,FLAGC5,FLAGD,FLAGEF,FLAGE,FLAGE1,FLAGF,FLAGF1,             10300002
     2FLAGG1,FLAGG2,FLAGH1,FLAGH2,TOPLRR,FLAGJ,FLAGK1,FLAGK2,FLAGL,     10310002
     3FLAGM,FLAGN,FLAGO,FLAGP,FLAGQ1,FLAGQ2,FLAGQ3,FCOUNT,JSTRAT,TOPEPA)10320002
        INTEGER FLAGA(120),FLAGB(120),FLAGC1(120),FLAGC2(120),FLAGC3(120),10330002
     1FLAGC4(120),FLAGD(120),FLAGEF(120),FLAGE(120),FLAGF(120),         10340002
     2FLAGG2(120),FLAGH1(120),FLAGH2(120),FLAGJ(120),FLAGK1(120),       10350002
     3FLAGK2(120),FLAGL(120),FLAGN(120),FLAGO(120),FLAGP(120),          10360002
     4FLAGQ1(120),FLAGQ2(120),FLAGQ3(120),FLAGC5(120),FCOUNT(120),      10370002
     5JSTRAT(120),FLAGE1(120)                                           10380002
        REAL A(628,36),TOPCA(120),TOPLRR(120),FLAGM(120),TOPEPA(120)    10390002
        CHARACTER FLAGG1(120)*1,FLAGF1(120)*1,FLAG(120)*18              10400002
        COMMON /RVAR/A                                                  10410002
        COMMON /CHTR/BFSCM,FLAG                                         10420002
        FLAG(I)=' '                                                     10430002
        FCOUNT(I)=0                                                     10440002
C CHECK FOR FLAG CONDITIONS A THRU Q                                    10450002
C FLAG C, QDR DISTRIBUTIONS REQUIRE ANOTHER SUBROUTINE                  10460002
C  FLAG A OCCURS WHEN MODEL DETECTS REINSPECTION HOURS BUT NO           10470002
C   CORRECTIVE ACTION OF AT LEAST TYPE B. MODEL LOOKS BACK ONE          10480002
C    MONTH TO SEE IF QDR WRITTEN.                                       10490002
C  FLAG A OCCURS WHEN THERE ARE LOTS REJECTED AND NO CORRECTIVE ACTIONS 10500010
C  OF AT LEAST TYPE B FOR TWO CONSECUTIVE MONTHS.FOR FACILITIES         10510010
C  THAT HAVE NO LOTS. THE EPA IS USED(UNITS HAVE BEEN REJECTED)         10520010
        IF (I .EQ. 1) GOTO 14                                           10530010
```

```
          IF((FLAGC2(I)+FLAGC3(I)+FLAGC4(I)+FLAGC5(I)) .GT. 0) GOTO 14      10540012
          IF((TOPLRR(I-1) .GT. 0.0) .AND. ((FLAGC2(I-1)+FLAGC3(I-1)+        10550010
         1FLAGC4(I-1)+FLAGC5(I-1)) .EQ. 0)) THEN                            10560010
           FLAG(I)(1:1)='A'                                                 10570013
           FCOUNT(I)=FCOUNT(I)+1                                            10580010
           GO TO 14                                                         10590010
         ELSE                                                               10600010
           IF(FLAGE(I-1) .GT. 0) GOTO 14                                    10610010
          IF(TOPEPA(I-1) .GT. 0.0) THEN                                     10620010
         IF((FLAGC2(I-1)+FLAGC3(I-1)+FLAGC4(I-1)+FLAGC5(I-1)) .GT. 0)       10630010
         1GOTO 14                                                           10640010
           FLAG(I)(1:1)='A'                                                 10650013
           FCOUNT(I)=FCOUNT(I)+1                                            10660010
          ENDIF                                                             10670010
        14  ENDIF                                                           10680010
C   FLAG B OCCURS WHEN MODEL DETECTS INTENSIFIED INSP HOURS BUT NO          10690002
C    CORRECTIVE ACTION OF AT LEAST TYPE B. MODEL LOOKS BACK ONE             10700002
C     MONTH TO SEE IF QDR WRITTEN.                                          10710002
      IF((FLAGB(I) .GT. 0) .AND. ((FLAGC2(I)+FLAGC3(I)+FLAGC4(I)+           10720010
     1FLAGC5(I)) .EQ. 0))  THEN                                             10730010
         IF(I .EQ. 1) GOTO 4                                                10740010
          IF((FLAGC2(I-1)+FLAGC3(I-1)+FLAGC4(I-1)+FLAGC5(I-1)) .GT. 0)      10750010
     1 GOTO 4                                                               10760010
        3    FLAG(I)(2:2)='B'                                               10770002
          FCOUNT(I)=FCOUNT(I)+1                                             10780002
        4  ENDIF                                                           10790002
C   FLAG C OCCURS WHEN CORRECTIVE ACTION DISTRIBUTION IS ABNORMAL           10800002
       CALL DISTR(I,TOPCA,FLAGC1,FLAGC2,FLAGC3,FLAGC4,FLAGC5,FCOUNT,        10810002
     1JSTRAT)                                                               10820002
C   FLAG D OCCURS WHEN MODEL DETECTS CORRECTIVE ACTION ABOVE TYPE A          10830010
C   WITHOUT MEETING HOURS.                                                  10840010
       IF((FLAGD(I) .EQ. 0) .AND. ((FLAGC2(I)+FLAGC3(I)+FLAGC4(I)+          10850010
      1FLAGC5(I)) .GT. 0)) THEN                                             10860010
         FLAG(I)(4:4)='D'                                                   10870002
         FCOUNT(I)=FCOUNT(I)+1                                              10880002
        ENDIF                                                               10890010
C   FLAG E OCCURS WHEN MODEL DETECTS A SHIPMENT WITHOUT PRODUCT             10900002
C   VERIFICATION INSPECTION HOURS.MODEL LOOKS BACK ONE                      10910002
C     MONTH TO SEE IF PVI OCCURED.                                          10920002
       IF((FLAGEF(I) .GT. 0) .AND. (FLAGE1(I) .EQ. 0)) THEN                 10930002
         IF(I .EQ. 1) GOTO 7                                                10940010
          IF(FLAGE1(I-1) .GT. 0) GOTO 7                                     10950002
        6    FLAG(I)(5:5)='E'                                               10960002
          FCOUNT(I)=FCOUNT(I)+1                                             10970002
        7  ENDIF                                                           10980002
C   FLAG F OCCURS WHEN MODEL DETECTS SHIPMENTS BUT NO VISITS AT             10990002
C    A NONRESIDENT FACILITY.                                                11000002
       IF((FLAGEF(I) .GT. 0) .AND. (FLAGF1(I) .EQ. 'N') .AND. (FLAGF(I)     11010002
      1 .EQ. 0)) THEN                                                       11020002
          FLAG(I)(6:6)='F'                                                  11030002
          FCOUNT(I)=FCOUNT(I)+1                                             11040002
       ENDIF                                                                11050002
C   FLAG G OCCURS UNDER THE FOLLOWING CIRCUMSTANCES                         11060002
C   1. MILQ OR MILI FACILITY AND                                            11070002
C   2. IF A RESIDENT FACILITY, THERE IS NO PROCESS EVALUTAION DURING        11080002
C      ANY MONTH - OR                                                       11090002
C   3. MILQ OR MILI FACILITY AND                                            11100002
C   4. IF A NONRESIDENT FACILITY, THERE IS EITHER NO PROCESS EVALUTAION     11110002
C      DURING THE FIRST MONTH OR THER IS NO PROCESS EVALUATION DURING AND   11120002
C      ANY SUBSEQUENT TWO CONSECUTIVE MONTHS.                               11130002
       IF((FLAGF1(I) .EQ. 'R') .AND. (FLAGG1(I) .NE. 'C') .AND.             11140002
      1 (FLAGG2(I) .EQ. 0)) THEN                                            11150002
        FLAG(I)(7:7)='G'                                                    11160002
        FCOUNT(I)=FCOUNT(I)+1                                               11170002
       ENDIF                                                                11180002
       IF((FLAGF1(I) .EQ. 'N') .AND. (FLAGG1(I) .NE. 'C') .AND.             11190002
```

```
     1 (FLAGG2(I) .EQ. 0) .AND. (I .EQ. 1)) THEN                        11200002
       FLAG(I)(7:7)='G'                                                 11210002
       FCOUNT(I)=FCOUNT(I)+1                                            11220002
      ENDIF                                                             11230002
      IF((FLAGF1(I) .EQ. 'N') .AND. (FLAGG1(I) .NE. 'C') .AND.          11240002
     1 (FLAGG2(I) .EQ. 0) .AND. (I .GT. 1)) THEN                        11250002
       IF(FLAGG2(I-1) .EQ. 0) THEN                                      11260002
         FLAG(I)(7:7)='G'                                               11270002
         FCOUNT(I)=FCOUNT(I)+1                                          11280002
       ENDIF                                                            11290002
      ENDIF                                                             11300002
C   FLAG H OCCURS WHEN MODEL DETECTS A CONTRACT RECEIVED LAST MONTH AND 11310002
C    NO PLANNING HOURS IN CURRENT MONTH OR PREVIOUS MONTH.              11320002
       IF(I .EQ. 1) GOTO 9                                              11330002
       IF((FLAGH1(I-1) .GT. 0) .AND. (FLAGH2(I-1) .EQ. 0)) THEN         11340002
          IF(FLAGH2(I) .GT. 0) GOTO 9                                   11350002
     8    FLAG(I)(8:8)='H'                                             11360002
         FCOUNT(I)=FCOUNT(I)+1                                          11370002
     9  ENDIF                                                          11380002
C   FLAG I OCCURS WHEN MODEL DETECTS LOTS REJECTED AND NO REINSPECTION  11390002
C    HOURS DURING MONTH OR NEXT MONTH IF NO LOTS ARE INSPECTED. UNIT    11400010
C    REJECTIONS WILL TRIGGER THE FLAG.                                  11410010
          IF(I .EQ. 1) GOTO 11                                          11420009
       IF(((TOPLRR(I-1) .GT. 0.0) .OR.((TOPEPA(I-1).GT. 0.0) .AND.      11430010
      1(FLAGE(I-1) .EQ. 0))) .AND. (FLAGA(I-1) .EQ. 0)) THEN            11440010
          IF(FLAGA(I) .GT. 0) GOTO 11                                   11450009
    10    FLAG(I)(9:9)='I'                                              11460002
          FCOUNT(I)=FCOUNT(I)+1                                         11470002
    11    ENDIF                                                         11480002
C   FLAG J OCCURS WHEN MDR DAYS TO CLOSE IS ABOVE NORMAL                11490002
       IF(REAL(FLAGJ(I)) .GT. A(JSTRAT(I),35)+2.0*(A(JSTRAT(I),36)+.001))11500002
      1 THEN                                                            11510002
          FLAG(I)(10:10)='J'                                            11520002
          FCOUNT(I)=FCOUNT(I)+1                                         11530002
       ENDIF                                                            11540002
C   FLAG K OCCURS WHEN NET QALI CHANGE IS ABOVE NORMAL                  11550002
C   NET QALI IS THE DIFFERENCE BETWEEN QALI RECEIVED AND QALI RESCINDED. 11560002
       IF(REAL(FLAGK1(I)-FLAGK2(I)).GT. A(JSTRAT(I),31)+1.0*(A(JSTRAT(I),11570002
      132)+.001)) THEN                                                  11580002
       FLAG(I)(11:11)='K'                                               11590002
          FCOUNT(I)=FCOUNT(I)+1                                         11600002
       ENDIF                                                            11610002
C   FLAG L OCCURS WHEN THE NUMBER OF NONQAR QDR ACTIONS IS ABOVE NORMAL. 11620002
C   LIMIT ON NONQAR CHANGED TO +1 SIGMA BASED ON DISCUSSION WITH RICH    11630002
C   ZERILLI AND RON DIPADOVA ON 29 JUNE 87. NO EXCEPTIONS TO GUIDANCE.  11640002
       IF(REAL(FLAGL(I)) .GT. A(JSTRAT(I),17)+1.0*(A(JSTRAT(I),18)+.001))11650002
      1 THEN                                                            11660002
          FLAG(I)(12:12)='L'                                            11670002
          FCOUNT(I)=FCOUNT(I)+1                                         11680002
       ENDIF                                                            11690002
C   FLAG M OCCURS WHEN THE SYSTEM INDICATOR IS ABOVE NORMAL.            11700002
       IF(FLAGM(I) .GT. 30.0) THEN                                      11710002
         FLAG(I)(13:13)='M'                                             11720002
         FCOUNT(I)=FCOUNT(I)+1                                          11730002
       ENDIF                                                            11740002
C   FLAG N OCCURS WHEN THE NUMBER OF PCO REQUESTS IS ABOVE NORMAL.      11750002
       IF(REAL(FLAGN(I)) .GT. A(JSTRAT(I),9)+1.0*(A(JSTRAT(I),10)+.001)) 11760002
      1 THEN                                                            11770002
         FLAG(I)(14:14)='N'                                             11780002
         FCOUNT(I)=FCOUNT(I)+1                                          11790002
       ENDIF                                                            11800002
C   FLAG O OCCURS WHEN THE NUMBER OF CAO REQUESTS IS ABOVE NORMAL.      11810002
       IF(REAL(FLAGO(I)) .GT. A(JSTRAT(I),11)+1.0*(A(JSTRAT(I),12)+.001))11820002
      1 THEN                                                            11830002
         FLAG(I)(15:15)='O'                                             11840002
         FCOUNT(I)=FCOUNT(I)+1                                          11850002
```

```
      ENDIF                                                              11860002
C  FLAG P OCCURS WHEN THE NUMBER OF ADMIN HOURS IS ABOVE NORMAL.          11870002
      IF(REAL(FLAGP(I)) .GT. A(JSTRAT(I),1)+2.0*(A(JSTRAT(I),2)+.001))   11880009
     1 THEN                                                              11890002
        FLAG(I)(16:16)='P'                                               11900002
        FCOUNT(I)=FCOUNT(I)+1                                            11910002
      ENDIF                                                              11920002
C  FLAG Q OCCURS WHEN THERE IS WORK NOT PERFORMED.                        11930002
      IF((FLAGQ1(I)+FLAGQ2(I)+FLAGQ3(I)).GT. 0) THEN                     11940002
        FLAG(I)(17:17)='Q'                                               11950002
        FCOUNT(I)=FCOUNT(I)+1                                            11960002
      ENDIF                                                              11970002
      RETURN                                                             11980011
      END                                                                11990011
C                                                                        12000002
C  *** THIS ROUTINE CHECKS FOR UNUSUAL CORRECTIVE ACTION DISTRIBUTIONS.  *12010002
C                                                                        12020002
      SUBROUTINE DISTR(I,TOPCA,FLAGC1,FLAGC2,FLAGC3,FLAGC4,FLAGC5,       12030002
     1FCOUNT,JSTRAT)                                                     12040002
      INTEGER FLAGC1(120),FLAGC2(120),FLAGC3(120),FLAGC4(120),           12050002
     1FLAGC5(120),FCOUNT(120),JSTRAT(120)                                12060002
      CHARACTER FLAG(120)*18                                             12070002
      REAL TOPCA(120),A(628,36)                                          12080002
      COMMON /RVAR/A                                                     12090002
      COMMON /CHTR/BFSCM,FLAG                                            12100002
      IF(TOPCA(I) .EQ. 0.0) GOTO 100                                     12110002
       IACNT=0                                                           12120002
       IBCNT=0                                                           12130002
       ICCNT=0                                                           12140002
       IDCNT=0                                                           12150002
       IECNT=0                                                           12160002
      DO 1 K=1,I                                                         12170002
       IACNT=IACNT+FLAGC1(K)                                             12180004
       IBCNT=IBCNT+FLAGC2(K)                                             12190004
       ICCNT=ICCNT+FLAGC3(K)                                             12200004
       IDCNT=IDCNT+FLAGC4(K)                                             12210004
       IECNT=IECNT+FLAGC5(K)                                             12220004
C  CHECK FOR OVER ESCALATION OF CORRECTIVE ACTION                        12230002
C    GENERAL                                                           12240002
C      IF((REAL(IBCNT)/REAL(K) .GT. A(JSTRAT(I),21)+1.0*A(JSTRAT(I),22))12250015
C     1.AND. (REAL(IACNT)/REAL(K) .LT. A(JSTRAT(I),19))) GOTO 2         12260015
C      IF((REAL(ICCNT)/REAL(K) .GT. A(JSTRAT(I),23)+1.0*A(JSTRAT(I),24))12270015
C     1.AND. (REAL(IBCNT)/REAL(K) .LT. A(JSTRAT(I),21))) GOTO 2         12280015
C      IF((REAL(IDCNT)/REAL(K) .GT. A(JSTRAT(I),25)+1.0*A(JSTRAT(I),26))12290015
C     1.AND. (REAL(ICCNT)/REAL(K) .LT. A(JSTRAT(I),23))) GOTO 2         12300015
C      IF((FLAGC3(K) .GT. 0) .AND. (IBCNT .EQ. 0)) GOTO 2                12310015
C      IF((FLAGC5(K) .GT. 0) .AND. (IBCNT .EQ. 0)) GOTO 2                12320015
C      IF((FLAGC4(K) .GT. 0) .AND. ((ICCNT+IECNT) .EQ. 0)) GOTO 2        12330015
C  CHECK FOR UNDER ESCALATION OF CORRECTIVE ACTION                       12340002
C  SINCE TYPICALLY THE STD DEV OF OOR DATA EXCEEDS THE MEAN, THE         12350002
C  DETECTION OF UNDER ESCALATION IS TRIGGERED BY BEING BELOW AVG.        12360002
      IF((REAL(IACNT)/REAL(K) .GT. A(JSTRAT(I),19)+1.0*A(JSTRAT(I),20))12370004
     1.AND. (REAL(IBCNT)/REAL(K) .LT. A(JSTRAT(I),21))) GOTO 2          12380004
      IF((REAL(IBCNT)/REAL(K) .GT. A(JSTRAT(I),21)+1.0*A(JSTRAT(I),22))12390004
     1.AND. (REAL(ICCNT)/REAL(K) .LT. A(JSTRAT(I),23))) GOTO 2          12400004
      IF((REAL(ICCNT)/REAL(K) .GT. A(JSTRAT(I),23)+1.0*A(JSTRAT(I),24))12410004
     1.AND. (REAL(IDCNT)/REAL(K) .LT. A(JSTRAT(I),25))) GOTO 2          12420004
      GO TO 1                                                            12430002
    2 FLAG(K)(3:3)='C'                                                   12440002
      IF (K .EQ. I) FCOUNT(I)=FCOUNT(I)+1                                12450002
    1 CONTINUE                                                           12460002
  100 RETURN                                                             12470002
      END                                                                12490002
C                                                                        12500002
C  *** THIS SUBROUTINE COMPUTES THE PROCESS EFFECTIVENESS BY DEDUCTING   12510002
C   POINTS FOR EACH RED FLAG GENERATED. ALSO THE PROCESS SCORE IS
```

```
C   COMBINED WITH THE PRODUCT SCORE TO COMPUTE AN OVERALL SCORE.          12520002
C                                                                        12530002
      SUBROUTINE SCORER(I,FCOUNT,TOPSCR,TP,JSTRAT)                       12540002
      INTEGER FCOUNT(120),JSTRAT(120)                                    12550002
      CHARACTER TP(120)*1                                                12560002
      REAL TOPSCR(12,120)                                                12570002
      TOPSCR(10,I)=100.0                                                 12580002
      IF(TP(I) .EQ. 'R') THEN                                            12590002
       IF(JSTRAT(I)/5*5 .EQ. JSTRAT(I)) THEN                             12600002
         IF(FCOUNT(I) .EQ. 1) TOPSCR(10,I)=90.0                          12610002
         IF(FCOUNT(I) .EQ. 2) TOPSCR(10,I)=80.0                          12620002
         IF(FCOUNT(I) .EQ. 3) TOPSCR(10,I)=65.0                          12630002
         IF(FCOUNT(I) .EQ. 4) TOPSCR(10,I)=50.0                          12640002
         IF(FCOUNT(I) .EQ. 5) TOPSCR(10,I)=35.0                          12650002
         IF(FCOUNT(I) .EQ. 6) TOPSCR(10,I)=20.0                          12660002
         IF(FCOUNT(I) .GE. 7) TOPSCR(10,I)= 5.0                          12670002
         IF(FCOUNT(I) .GE. 8) TOPSCR(10,I)= 0.0                          12680002
        GO TO 10                                                         12690002
       ENDIF                                                             12700002
       IF(FCOUNT(I) .EQ. 1) TOPSCR(10,I)=90.0                            12710002
       IF(FCOUNT(I)  EQ  2) TOPSCR(10,I)=75.0                            12720002
       IF(FCOUNT(I) .EQ. 3) TOPSCR(10,I)=60.0                            12730002
       IF(FCOUNT(I) .EQ. 4) TOPSCR(10,I)=45.0                            12740002
       IF(FCOUNT(I) .EQ. 5) TOPSCR(10,I)=30.0                            12750002
       IF(FCOUNT(I) .EQ. 6) TOPSCR(10,I)=15.0                            12760002
       IF(FCOUNT(I) .GE. 7) TOPSCR(10,I)=0.0                             12770002
      ELSE                                                               12780002
       IF(FCOUNT(I) .EQ. 1) TOPSCR(10,I)=80.0                            12790002
       IF(FCOUNT(I) .EQ. 2) TOPSCR(10,I)=60.0                            12800002
       IF(FCOUNT(I) .EQ. 3) TOPSCR(10,I)=40.0                            12810002
       IF(FCOUNT(I) .EQ. 4) TOPSCR(10,I)=20.0                            12820002
       IF(FCOUNT(I) .GE. 5) TOPSCR(10,I)=0.0                             12830002
10    ENDIF                                                              12840002
      TOPWT=.4                                                           12850002
      COAPWT=.6                                                          12860002
      TOPSCR(11,I)=TOPSCR(9,I)*TOPWT+TOPSCR(10,I)*COAPWT                 12870002
       IF(I .EQ. 1) THEN                                                 12880002
        TOPSCR(12,I)=50.0                                                12890002
       ELSE                                                              12900002
        TOPSCR(12,I)=TOPSCR(11,I-1)                                      12910002
       ENDIF                                                             12920002
      RETURN                                                             12930002
      END                                                                12940002
```

Appendix F

Group Identification Number Conversion Tables

Certain commodities were combined. The model does not distinguish commodity differences for the following:

Aircraft (A) and Missiles/Space (X)

Munitions (B) and Weapons (W)

Marine (D) and Automotive (V)

Electrical (E), Electronic Systems (K) and Electronics (L)

General (G) and Service (S)

Chemical (H) and Petroleum (P)

In the conversion table on the next page, the primary group number reflects the initial, unconstrained grouping described in Appendix E-9 through E-10. The secondary group number reflects the above combinations per Appendix E-11.

12-11-85

RESIDENT FACILITY CONVERSION TABLE

	PRIM-ARY GROUP NO	SECON-ARY GROUP NO
AQ0 2	1	1
AQ3 7	2	2
AQ8 20	3	3
AQ	4	4
AQ20+	5	5
AI0 2	6	6
AI3 7	7	7
AI8 20	8	8
AI	9	9
AI20+	10	10
BQ0 2	11	11
BQ3 7	12	12
BQ8 20	13	13
BQ	14	14
BQ20+	15	15
BI0 2	16	16
BI3 7	17	17
BI8 20	18	18
BI	19	19
BI20+	20	20
CI0 2	21	21
CI3 7	22	22
CI8 20	23	23
CI	24	24
CI20+	25	25
CQ0 2	26	26
CQ3 7	27	27
CQ8 20	28	28
CQ	29	29
CQ20+	30	30
DQ0 2	31	31
DQ3 7	32	32
DQ8 20	33	33
DQ	34	34
DQ20+	35	35
DI0 2	36	36
DI3 7	37	37
DI8 20	38	38
DI	39	39
DI20+	40	40
EQ0 2	41	41
EQ3 7	42	42
EQ8 20	43	43
EQ	44	44
EQ20+	45	45

	PRIM-ARY GROUP NO	SECON-ARY GROUP NO
EQ0 2	46	196
EQ3 7	47	197
EQ8 20	48	198
EQ	49	199
EQ20+	50	200
EI0 2	51	201
EI3 7	52	202
EI8 20	53	203
EI	54	204
EI20+	55	205
FQ0 2	56	206
FQ3 7	57	207
FQ8 20	58	208
FQ	59	209
FQ20+	60	210
GQ0 2	61	121
GQ3 7	62	122
GQ8 20	63	123
GQ	64	124
GQ20+	65	125
GI0 2	66	126
GI3 7	67	127
GI8 20	68	128
GI	69	129
GI20+	70	130
HQ0 2	71	131
HQ3 7	72	132
HQ8 20	73	133
HQ	74	134
HQ20+	75	135
HI0 2	76	76
HI3 7	77	77
HI8 20	78	78
HI	79	79
HI20+	80	80
JI0 2	81	81
JI3 7	82	82
JI8 20	83	83
JI	84	84
JI20+	85	85
JQ0 2	86	86
JQ3 7	87	87
JQ8 20	88	88
JQ	89	89
JQ20+	90	90

	PRIM-ARY GROUP NO	SECON-ARY GROUP NO
KQ0 2	91	166
KQ3 7	92	167
KQ8 20	93	168
KQ	94	169
KQ20+	95	170
KI0 2	96	171
KI3 7	97	172
KI8 20	98	173
KI	99	174
KI20+	100	175
LQ0 2	101	176
LQ3 7	102	177
LQ8 20	103	178
LQ	104	179
LQ20+	105	180
MQ0 2	106	121
MQ3 7	107	122
MQ8 20	108	123
MQ	109	124
MQ20+	110	125
MI0 2	111	126
MI3 7	112	127
MI8 20	113	128
MI	114	129
MI20+	115	130
NQ0 2	116	131
NQ3 7	117	132
NQ8 20	118	133
NQ	119	134
NQ20+	120	135
OQ0 2	121	121
OQ3 7	122	122
OQ8 20	123	123
OQ	124	124
OQ20+	125	125
OI0 2	126	126
OI3 7	127	127
OI8 20	128	128
OI	129	129
OI20+	130	130
PQ0 2	131	131
PQ3 7	132	132
PQ8 20	133	133
PQ	134	134
PQ20+	135	135

	PRIM-ARY GROUP NO	SECON-ARY GROUP NO
QQ0 2	136	136
QQ3 7	137	137
QQ8 20	138	138
QQ	139	139
QQ20+	140	140
QI0 2	141	141
QI3 7	142	142
QI8 20	143	143
QI	144	144
QI20+	145	145
RQ0 2	146	146
RQ3 7	147	147
RQ8 20	148	148
RQ	149	149
RQ20+	150	150
RI0 2	151	151
RI3 7	152	152
RI8 20	153	153
RI	154	154
RI20+	155	155
SI0 2	156	156
SI3 7	157	157
SI8 20	158	158
SI	159	159
SI20+	160	160
TQ0 2	161	161
TQ3 7	162	162
TQ8 20	163	163
TQ	164	164
TQ20+	165	165
UQ0 2	166	166
UQ3 7	167	167
UQ8 20	168	168
UQ	169	169
UQ20+	170	170
UI0 2	171	171
UI3 7	172	172
UI8 20	173	173
UI	174	174
UI20+	175	175
VQ0 2	176	176
VQ3 7	177	177
VQ8 20	178	178
VQ	179	179
VQ20+	180	180

	PRIM-ARY GROUP NO	SECON-ARY GROUP NO
SQ0 2	181	76
SQ3 7	182	77
SQ8 20	183	78
SQ	184	79
SQ20+	185	80
SI0 2	186	81
SI3 7	187	82
SI8 20	188	83
SI	189	84
SI20+	190	85
SQ0 2	191	86
SQ3 7	192	87
SQ8 20	193	88
SQ	194	89
SQ20+	195	90
VQ0 2	196	196
VQ3 7	197	197
VQ8 20	198	198
VQ	199	199
VQ20+	200	200
VI0 2	201	201
VI3 7	202	202
VI8 20	203	203
VI	204	204
VI20+	205	205
WQ0 2	206	206
WQ3 7	207	207
WQ8 20	208	208
WQ	209	209
WQ20+	210	210
WQ0 2	211	16
WQ3 7	212	17
WQ8 20	213	18
WQ	214	19
WQ20+	215	20
WI0 2	216	21
WI3 7	217	22
WI8 20	218	23
WI	219	24
WI20+	220	25
WQ0 2	221	26
WQ3 7	222	27
WQ8 20	223	28
WQ	224	29
WQ20+	225	30

	PRIM-ARY GROUP NO	SECON-ARY GROUP NO
XQ0 2	226	1
XQ3 7	227	2
XQ8 20	228	3
XQ	229	4
XQ20+	230	5
XI0 2	231	6
XI3 7	232	7
XI8 20	233	8
XI	234	9
XI20+	235	10
YQ0 2	236	11
YQ3 7	237	12
YQ8 20	238	13
YQ	239	14
YQ20+	240	15

FIRST POSITION IS COMMODITY
SECOND POSITION IS CA PROVISION
THIRD POSITION IS MIN NO OF DAYS
FOURTH POSITION IS MAX NO OF DAYS

Nonresident Facilities

A more complex method was needed to combine nonresident facilities to eliminate voids in the data. The following criteria are listed in order of precedence:

1. MILQ Facilities. The second alpha of the commodity code is ignored. Thus all nonresident, MILQ facilities are combined based only on primary commodity code.

2. MILI Facilities. The second alpha of the commodity code is ignored for Chemical, Nuclear and Service commodities.

3. Standard Inspection. The second alpha for Nuclear commodities is ignored. For example, for nonresident Nuclear facilities there are only three actual groups (MILQ, MILI and Standard Inspection).

4. Special Cases. Combining commodities does not imply crossing QA provision boundaries. MILI and standard inspection facilities are never grouped together.

a. Marine Repair commodities (D2) were grouped with nonresident maintenance facilities even if Operation Type Code is not "C".

b. Munitions B1 (Conventional Explosives) and B2 (all ammunition) were combined.

c. Chemical H1 (military chemical agents) and H2 (Military biological agents) were combined for standard inspection facilities.

d. Facilities were combined for B3 (fuses), B4 (Munitions metal parts), B5 (Munitions, Miscellaneous) and W6 (Miscellaneous Weapons).

e. Facilities were combined for Clothing and Textiles, C3 (Footwear and leather goods), C5 (Embroidered Insignia) and C6 (Miscellaneous).

f. Electronics L1 (Electron Tubes) and I.5 (Miscellaneous) were combined.

g. Petroleum P1 (Petroleum Transportation and Storage Services), P4 (Crude Oil), P6 (Other fuels) and P7 (Petroleum "Into Planes" Service) were combined.

h. Weapons W2 (Artillery), W4 (Rocket Launchers and Auxiliary Support Equipment) and W5 (Auxiliary Support Equipment and Weapon Systems Components) were combined.

i. Missiles and Space X1 (Guided Missiles and Components), X2 (Space Vehicles and Components including Satellites) and A4 (Aircraft/Airframe Structured Components and Subassemblies) were combined.

j. Missiles and Space X4 (Ground Support or Handling Equipment), X5 (Miscellaneous) and Aircraft A3 (Aircraft Electronic Components/Subassemblies) and A7 (Miscellaneous) were combined.

k. Missiles and Space X3 (Instrumentatior/Simulators and Control Equipment) and Aircraft A6 (Aircraft Simulators) were combined.

l. Weapons W3 (Armored Vehicles and Weapon Carriers) and V1 (Vehicles and Trailers) were combined.

m. Electrical E2 (Power and Distribution Equipment) and E6 (Miscellaneous) were combined.

The above combinations are reflected in the conversion table on the next page. The code that creates this constrained grouping is documented in Appendix E-11 through E-12.

12-11-86

NONRESIDENT FACILITY CONVERSION TABLE

PRIM-SECONDARY GROUP NO	SECONDARY GROUP NO
A10 241	241
A11 242	242
A10 243	243
A20 244	241
A21 245	245
A20 246	246
A30 247	241
A31 248	260
A30 249	261
A40 250	241
A41 251	251
A40 252	252
A50 253	625
A51 254	625
A50 255	625
A60 256	241
A61 257	257
A60 258	258
A70 259	241
A71 260	260
A70 261	261
A80 262	241
A81 263	263
A80 264	264
B10 265	265
B11 266	266
B10 267	267
B20 268	265
B21 269	266
B20 270	267
B30 271	265
B31 272	278
B30 273	279
B40 274	265
B41 275	278
B40 276	279
B50 277	265
B51 278	278
B50 279	279
B60 280	265
B61 281	281
B60 282	282
B70 283	265
B71 284	284
B70 285	285
B80 286	265
B81 287	287
B80 288	288

PRIM-SECONDARY GROUP NO	SECONDARY GROUP NO
C10 289	289
C11 290	290
C10 291	291
C20 292	289
C21 293	293
C20 294	294
C30 295	289
C31 296	305
C30 297	306
C40 298	289
C41 299	299
C40 300	300
C50 301	289
C51 302	305
C50 303	306
C60 304	289
C61 305	305
C60 306	306
C70 307	289
C71 308	308
C70 309	309
C80 310	289
C81 311	311
C80 312	312
B10 313	313
B11 314	314
B10 315	315
B20 316	625
B21 317	625
B20 318	625
B30 319	313
B31 320	320
B30 321	321
B40 322	313
B41 323	323
B40 324	324
B50 325	313
B51 326	326
B50 327	327
B60 328	313
B61 329	329
B60 330	330
B70 331	313
B71 332	332
B70 333	333
B80 334	313
B81 335	335
B80 336	336

PRIM-SECONDARY GROUP NO	SECONDARY GROUP NO
E10 337	337
E11 338	338
E10 339	339
E20 340	337
E21 341	353
E20 342	354
E30 343	337
E31 344	344
E30 345	345
E40 346	337
E41 347	347
E40 348	348
E50 349	337
E51 350	350
E50 351	351
E60 352	337
E61 353	353
E60 354	354
E70 355	337
E71 356	356
E70 357	357
E80 358	337
E81 359	359
E80 360	360
G10 361	361
G11 362	362
G10 363	363
G20 364	361
G21 365	365
G20 366	366
G30 367	361
G31 368	368
G30 369	369
G40 370	361
G41 371	371
G40 372	372
G50 373	361
G51 374	374
G50 375	375
G60 376	361
G61 377	377
G60 378	378
G70 379	361
G71 380	380
G70 381	381
G80 382	361
G81 383	383
G80 384	384

PRIM-SECONDARY GROUP NO	SECONDARY GROUP NO
H10 385	385
H11 386	386
H10 387	387
H20 388	385
H21 389	386
H20 390	387
H30 391	385
H31 392	386
H30 393	393
H40 394	385
H41 395	386
H40 396	396
H50 397	385
H51 398	386
H50 399	399
H60 400	385
H61 401	386
H60 402	402
H70 403	385
H71 404	386
H70 405	405
H80 406	385
H81 407	386
H80 408	408
K10 409	409
K11 410	410
K10 411	411
K20 412	409
K21 413	413
K20 414	414
K30 415	409
K31 416	416
K30 417	417
K40 418	409
K41 419	419
K40 420	420
K50 421	409
K51 422	422
K50 423	423
K60 424	409
K61 425	425
K60 426	426
K70 427	409
K71 428	428
K70 429	429
K80 430	409
K81 431	431
K80 432	432

PRIM-SECONDARY GROUP NO	SECONDARY GROUP NO
L10 433	433
L11 434	446
L10 435	447
L20 436	433
L21 437	437
L20 438	438
L30 439	433
L31 440	440
L30 441	441
L40 442	433
L41 443	443
L40 444	444
L50 445	433
L51 446	446
L50 447	447
L60 448	433
L61 449	449
L60 450	450
L70 451	433
L71 452	452
L70 453	453
L80 454	433
L81 455	455
L80 456	456
M10 457	457
M11 458	458
M10 459	459
M20 460	457
M21 461	461
M20 462	462
M30 463	457
M31 464	464
M30 465	465
M40 466	457
M41 467	467
M40 468	468
M50 469	457
M51 470	470
M50 471	471
M60 472	457
M61 473	473
M60 474	474
M70 475	457
M71 476	476
M70 477	477
M80 478	457
M81 479	479
M80 480	480

PRIM-SECONDARY GROUP NO	SECONDARY GROUP NO
N10 481	481
N11 482	482
N10 483	483
N20 484	481
N21 485	482
N20 486	483
N30 487	481
N31 488	482
N30 489	483
N40 490	481
N41 491	482
N40 492	483
N50 493	481
N51 494	482
N50 495	483
N60 496	481
N61 497	482
N60 498	483
N70 499	481
N71 500	482
N70 501	483
N80 502	481
N81 503	482
N80 504	483
P10 505	505
P11 506	524
P10 507	525
P20 508	505
P21 509	509
P20 510	510
P30 511	505
P31 512	512
P30 513	513
P40 514	505
P41 515	524
P40 516	525
P50 517	505
P51 518	518
P50 519	519
P60 520	505
P61 521	524
P60 522	525
P70 523	505
P71 524	524
P70 525	525
P80 526	505
P81 527	527
P80 528	528

PRIM-SECONDARY GROUP NO	SECONDARY GROUP NO
S10 529	529
S11 530	530
S10 531	531
S20 532	529
S21 533	530
S20 534	534
S30 535	529
S31 536	530
S30 537	537
S40 538	529
S41 539	530
S40 540	540
S50 541	529
S51 542	530
S50 543	543
S60 544	529
S61 545	530
S60 546	546
S70 547	529
S71 548	530
S70 549	549
S80 550	529
S81 551	530
S80 552	552
V10 553	553
V11 554	554
V10 555	555
V20 556	553
V21 557	557
V20 558	558
V30 559	553
V31 560	560
V30 561	561
V40 562	553
V41 563	563
V40 564	564
V50 565	553
V51 566	566
V50 567	567
V60 568	553
V61 569	569
V60 570	570
V70 571	553
V71 572	572
V70 573	573
V80 574	553
V81 575	575
V80 576	576

PRIM-SECONDARY GROUP NO	SECONDARY GROUP NO
W10 577	577
W11 578	578
W10 579	579
W20 580	577
W21 581	581
W20 582	582
W30 583	577
W31 584	464
W30 585	465
W40 586	577
W41 587	581
W40 588	582
W50 589	577
W51 590	581
W50 591	582
W60 592	577
W61 593	278
W60 594	279
W70 595	577
W71 596	596
W70 597	597
W80 598	577
W81 599	599
W80 600	600
X10 601	601
X11 602	251
X10 603	252
X20 604	601
X21 605	251
X20 606	252
X30 607	601
X31 608	257
X30 609	258
X40 610	601
X41 611	260
X40 612	261
X50 613	601
X51 614	260
X50 615	261
X60 616	601
X61 617	617
X60 618	618
X70 619	601
X71 620	620
X70 621	621
X80 622	601
X81 623	623
X80 624	624

FIRST POSITION IS COMMODITY
SECOND POSITION IS COMMODITY
THIRD POSITION IS CA REGULATION

Appendix G

Red Flags

Flag 'A.' "Lots or Units Rejected without Corrective Action." If, in the preceding month of record, there was a lot rejected and there has been no corrective action taken of at least type B (written) during the prior month or the current month, the 'A' flag is generated. If there was no lot inspected, the model performs a similar check on units inspected and rejected during the previous month. Thus for a facility that did not report lots, units rejected is used in lieu of lots rejected. If there is a reject, the QAR has the current month or the next month to issue a type B or higher QDR to avoid the 'A' flag in the next month. Note: preceding month of record means the previous month or the previous visit whichever is greater. For a non-resident facility, the preceding month of record could be several months in the past. Usually, however, the preceding month of record is the prior month.

Flag 'B.' "Intensified Inspection Without Corrective Action." When intensified inspection hours are reported, previous rejections are implied. Therefore, corrective actions of a least level B are required. If the model fails to find at least one type B QDR in the month of intensified inspection or the prior month (of record), Flag 'B' occurs. Caution: This flag could be unfairly generated when the contractor takes a very long time to correct causes of rejection and resubmit items for reinspection.

Flag 'C.' "Abnormal Corrective Action Distribution." For a given peer group, a certain distribution of corrective actions, by type or level, is expected. If there are many low level type QDR's and few high level types, this may indicate that the QAR is not properly escalating QDR's. When, relative to peers, a facility tends to have lower level QDR's which are at least one standard deviation above average yet is below average for the next higher level of QDR, Flag 'C' occurs as follows:

1. The average number of Type A QDR's exceeds plus one standard deviation from normal and the average number of Type B QDR's is below normal.

2. The average number of Type B QDR's exceeds plus one standard deviation from normal and the average number of Type C QDR's is below normal.

3. The average number of Type C QDR's is above one standard deviation from the group average and the average number of Type D QDR's is below normal. Note: Since Type C and D QDR's are relatively rare, this flag will almost always appear whenever Type C occurs and Type D does not also occur unless the model is run for several months at a time. Note: Underescalation flags tend to recur because of the averaging process. A facility may have to alter its QDR pattern for several months to restore a proper balance and cause the flag to cease.

Flag 'D.' "Corrective Actions Without Meeting with Contractor." QAR's are required to meet periodically with top management of the contractor to discuss CQAP issues. If the QAR issues a corrective action of at least type B, the QAR should meet with the contractor during that month. If there is at least one type B or above QDR and there are no meeting hours reported during the

month of the QDR, Flag 'D' occurs. Note: There is no tolerance given for timing of the two events. It is conceivable that the QDR could be written on the last day of the month and the meeting conducted on the first day of the next month. However QAR's are suppose to meet with the plant manager "frequently" per DLAM 8200.1 regardless of QDR's.

Flag 'E.' "Shipments without Product Verification Inspection." If product is shipped, some inspection should be made regardless of the extensive use of Procedure Evaluation (PE) and Product Oriented Procedure Evaluation (POPE). Some QAR's mistakenly believe that PE and POPE can eliminate the need completely for Product Verification Inspection (PVI). The model will issue an 'E' Flag during any month a shipment is made and there have been no inspection hours reported during that month or the prior month.

Flag 'F.' "Shipments without Visits-Nonresident." If a nonresident facility ships product during a given month and there is no record of a corresponding visit by the QAR, a Flag 'F' appears. A combination of zero visit count and non-zero shipment count causes a nonresident facility to receive an 'F' flag. Note: Many QAR's do not currently count visits whenever work is done at the facility that is the QAR's duty-station. To prevent this flag from erroneously occuring, visits must be counted whenever a nonresident performs CQA within his/her duty station. DLAM 8200.2 requires these visits to be counted but many QAR's fail to do so.

Flag 'G.' "No Procedure Evaluation at MILI or MILQ Facility." If, for a nonresident facility, in any two consecutive months or visits, whichever is greater, no PE hours are recorded against a MILI or MILQ facility, Flag 'G' is generated. Resident MILI or MILQ facilities must have PE hours each month to avoid the flag. This flag occurs in over one in five facilities each month. Note: This flag could occur in active nonresident facilities that have a small number of special processes that have 90, 120, 180 day intervals for PE. Also, this flag could be generated for active MILI and MILQ facilities that have only standard inspection contracts for a period of time.

Flag 'H.' "Contracts Received without Planning." If a contract was received during the prior month and no planning hours were reported for the prior month and the current month, Flag 'H' occurs.

Flag 'I.' "Lots Rejected without Reinspection." If at least one lot was rejected the prior month, and there has been no reinspection hours reported for the prior month and the current month, Flag 'I' is generated for the current month. If no lots were inspected, unit rejections are evaluated in lieu of lots. Note: This flag could be caused by contractor delays in submitting lots or units for reinspection.

Flag 'J.' "Excessive Days to Close a Materiel Deficiency Report." If the average days to close valid MDR's received during a month exceed the group average by two standard deviations, Flag 'J' will be generated. Only MDR's deemed "valid" per the criteria described in Appendix B, page B-2 are analyzed. Thus open MDR's that are also excessive are not flagged. Depending on the group, flags tend to occur when the days to close approach approximately 100 days $\pm$ 30 days.

Flag 'K.' "Excessive Net Quality Assurance Letters of Instruction (QALI)." High numbers of QALI's on hand are an indication of lack of Principal Contracting Officer (PCO) confidence in DCAS CQA. If the QAR convinces the PCO to rescind a QALI, an indication of confidence is inferred. The model computes a variable called net QALI which subtracts QALI's rescinded from QALI on hand. When this variable is excessive, compared to peers, Flag 'K' occurs.

The criteria for Flag 'K' is when net QALI is more than one standard deviation from average. Note: The criteria chosen to determine the number of standard deviations allowed before flagging was to use one standard deviation if the standard deviation was more than the average. If the group averages tended to exceed the group standard deviations, then two standard deviation limits were used.

Flag 'L.' "Non QAR QDR Actions." If the number of corrective actions written by someone other than the QAR is above average by one standard deviation, Flag 'L' occurs. Note: These events rarely occur or are seldom reported. Thus the average and standard deviations for non QAR QDR's are very low. A single occurrence is sufficient to generate this flag for all groups but one using the existing Master Data File.

Flag 'M.' "Excessive System Indicator (SI)." High SI values indicate that the QAR is spending a large percentage of the time on adverse actions such as QDR, MDR, Material Review Board (MRB) etc. The 'M' Flag occurs whenever SI exceeds 30%. Note: This flag should be group dependent but reprogramming would be extensive. This action has been deferred to planned future model enhancements.

Flag 'N.' "Excessive Principal Contracting Officer Request." Frequent and excessive PCO requests are a sign of lack of confidence by the PCO. When the number of requests exceeds the group average by one standard deviation, the 'N' flag is assigned.

Flag 'O.' "Excessive Contract Administrative Office (CAO) Requests." Same as Flag 'N' concerning the number of CAO requests received during the month of record.

Flag 'P.' "Excessive Administrative Hours." Administrative hours is a "catch all" category of time reported to pick up all time not reported in other data elements. If the number of hours exceeds the group average by two standard deviations, Flag P occurs. Note: Administrative hours are a category of

uncertainty with potential changes in definition. For example, Quality Data Evaluation (QDE) has been reported as administrative time but plans exist to create a new data element for QDE. Significant changes in any data element definition will cause problems with flag generation, requiring model update.

Flag 'Q.' "Work not Performed." If, in any month, there is any work not performed reported, the 'Q' flag will appear. The three activities scanned for non-zero data are:

1. Number of product verification inspections not performed.

2. Number of procedure evaluation locations not evaluated.

3. Number of procedure evaluation elements not evaluated.

Appendix H

Product Effectiveness

DEFINITIONS OF PRODUCT INDICATORS

Estimated Process Average (EPA). EPA is defined as the ratio of units rejected to units inspected. It is reported in QAMIS as a percentage.

Lot Rejection Ratio (LRR). LRR is defined as the ratio of lots rejected to lots inspected. All shipments are not inspected and reported in lots. Thus it is possible to have a positive EPA and a zero LRR. Also because of Acceptable Quality Levels allowed in MIL-STD-105D, it is possible to reject units yet accept the lot. Thus LRR and EPA are related but not totally dependent indicators.

Material Review Board (MRB). MRB actions are the number of actions reviewed by the QAR. MRB is a mechanism for the contractor to obtain approval of minor non-conformances (not affecting form, fit or function).

Waivers and Deviations (W/D). All waivers and deviations reviewed by the QAR are reported. A major cause (but not the only reason) for waivers and deviations is that the contractor is unable to conform to the existing specifications. The contractor can seek relief via the waiver/deviation process.

Engineering Change Proposal (ECP). ECP actions are a request to change the specification. ECP actions reviewed by the QAR are reported. Some ECPs are caused by the contractor's inability to meet specifications.

Corrective Actions (CA). The number of Quality Deficiency Reports generated by the QAR is reported by type. There are five types of QDRs ranging from on-the-spot verbal, Type A, to termination of CQAP, type D. An artificial indicator was created to produce a CA value by taking the weighted value of the QDR's by type. Weighting factors were produced by the SAG using SPAN methodology.

1. Type A - Weight = 3

2. Type B - Weight = 10. The SAG felt that a Type B (written) corrective action was 3 1/3 times more serious than a Type A QDR. Thus three Type A QDR's would almost generate the same CA effect as a Type B.

3. Type C - Weight = 30. The SAG felt that a Type C (escalated to higher levels) was 3 times more serious than a Type B QDR. Thus 3 Type B QDR's would generate the same effect as a Type C.

4. Type D - Weight = 60. The SAG felt that two Type C QDRs should carry the same weight as a type D QDR.

5. Type E - Weight = 30. The SAG felt that a Type C and a Type E (subcontractor corrective action) were roughly equivalent.

Materiel Deficiency Reports (MDR). The weighted MDR counts discussed in paragraph II B1 represent the number of valid MDRs (or customer complaints).

NORMALIZATION OF PRODUCT INDICATORS

Indicator values. It was observed that the distribution of product indicator values tended to follow roughly the shape of the exponential distribution rather than a normal, bell shaped distribution. Counts and ratios were normalized by computing a value called a Z value where:

$$Z = \log \frac{\text{indicator count or value}}{\text{avg indicator count or value}}$$

The average indicator count or value is the group average found in the Master Data File discussed in paragraph II D3.

Z values less than -3.0 or greater than +3.0 were clipped to -3.0 and +3.0 respectively.

For example, suppose a facility in group number 22 had an EPA of 15% in a certain month. The average EPA for similar facilities is 5.5%. The Z-value for that indicator value is computed to be log (15/5.5) = +1.0033. (NOTE: Log is the natural logarithm.) Thus a value of 15% can be expressed as being approximately one standard deviation above normal.

Indicator Rates. The rate of change was computed by breaking the total history of indicator values into two equal groups from the midpoint of the time series of indicator values. If there are more than 12 months of data, the model ignores the earlier data. If there is an even number of monthly data, the break is clean at the n/2th month. If there is an odd number of monthly data points, the middle month is excluded as belonging to neither group. The average indicator value is computed for the first half's sequence and also for the second half. The rate is computed as follows:

$$Z = 3 * \left[\frac{\text{Avg value for 2nd half of time series}}{\text{Avg value of 1st half of time series}} \right] - 3$$

Z values of less than -3 and greater than +3 are clipped to -3 and +3 respectively.

For example, suppose a facility had seven observations for MRB counts up to a certain month. The time series of MRB counts was 6, 10, 14, 9, 10, 5, 3. The Z value is $3\left(\frac{10+5+3}{6+10+14}\right) - 3 = -1.2$. Thus the indicator is decreasing at a moderate rate.

CRITERIA FOR IDENTIFICATION OF PROBLEM CONTRACTOR

Criteria

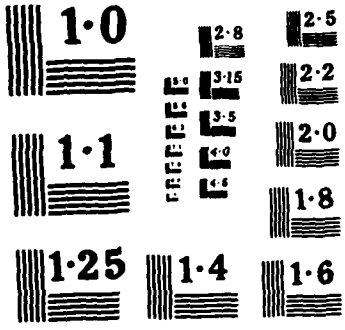
1. Contractor is on the Alert list* for any cause.
2. Contractor is more than one standard deviation above average in MDR count.
3. Contractor has received a Type C corrective action.
4. Contractor has received a Type E corrective action.
5. Contractor once received a Type D correction action.

* The Alert list used by QUEST must not contain more than 2000 FSCMs. QUEST will not execute.

The first criteria listed above will be checked against the current Alert list, regardless of the month of record. For the second criteria, the model looks at the current month of record and prior month for nonresident facilities but at the entire history for resident facilities. Thus nonresident facilities may jump back and forth from "problem" to "normal" facilities as MDR's are received. Resident facilities with above average MDR counts may take longer to remove the label of "problem" and are more stable. The third and fourth criteria are also based on the month of record occurence and will also be "on" or "off" month to month. The last criteria is either always "on" or "off." The only mechanism to remove the effect of a Type D corrective action is to wait until the cut-off parameter takes effect (see Appendix E-2). Depending on the criteria that activates the identification, a facility may be a "problem" facility one month and "normal" the following month. This is especially true for nonresident facilities.

If a nonresident facility meets any of the criteria shown, it is labeled "Problem Facility." If a resident facility meets two or more of the criteria, it is labeled "Problem Facility." Documentation is found in Appendix E-12 through E-13.

AD-A202 017 ANALYSIS OF QUALITY ASSURANCE (QA) EFFECTIVENESS(U) 2/2
DEFENSE LOGISTICS AGENCY ALEXANDRIA VA OPERATIONS
RESEARCH AND ECONOMIC ANALY SIS OFFICE P E GROVER
UNCLASSIFIED JUL 88 F/G 12/4 NL


1·0
1·1
1·25
1·4
1·6
2·8
3·15
3·5
2·5
2·2
2·0
1·8

Z VALUES FOR IDEAL/NEGATIVE IDEAL CASES

	Ideal Facility		Neg Ideal Facility	
	Normal	Problem	Normal	Problem
EPA Ratio	-3		+3	
EPA Rate	-3	-3	+3	+3
LRR Ratio	-3		+3	
LRR Rate	-3	-3	+3	+3
MRB Count	-3		+3	
MRB Rate	-3	-3	+3	+3
W/D Count	-3		+3	
W/D Rate	-3	-3	+3	+3
ECP Count	-3		+3	
ECP Rate	-3	-3	+3	+3
CA Count	-3	+3	+3	-3
CA Rate	-3		+3	
MDR Count	-3	-3	+3	+3
MDR Rate	-3		+3	

For a given month's data, each product indicator is scored by computing the distances the normalized values and rates are from the ideal and negative ideal points. For normal facilities, QUEST uses the city-block or rectilinear distance formula described by Hwang [2] for computing distances. Distances for problem facilities are one-dimensional. For example, Facility XYZ, a normal facility, has Z values of -2.0 and +.5 for a certain indicator in given month. The distance to the negative ideal point is (3-(-2)) plus (3-.5) = 7.5. The distance to the ideal point is (-2-3) plus (.5-(-3)) = 4.5. The score is the ratio of the distance from negative ideal divided by the total distance or $\left[\frac{7.5}{7.5+4.5}\right]$ = .625 or 62.5%.

COMPUTATION OF PRODUCT EFFECTIVENESS

$$\text{Product Score} = \alpha_j \sqrt{\frac{\sum_i wt^2_i \times d^{2-}_i}{\sqrt{\sum_i wt^2_i d^{2-}_i} + \sqrt{\sum_i wt^2_i d^{2+}}}}$$

where $i = 1.....14$

or $i = 1.....7$

wt_i = weight factor from SAG

d^{2-}_i = distance squared from negative ideal

d^{2+}_i = distance squared from positive ideal

j = Knob for problem facilities = .79

(Knob for normal facilities = 1.00)

A "Knob" of 79% was applied to the product score of problem facilities to reflect the expected lower product quality achievable under the best of conditions. If the contractor is "problem," the highest score achievable is 79%. The QAR can only achieve higher scores by getting the contractor reclassified by QUEST as "normal." This can be done by reducing MDRs, eliminating the need for type C and E QDRs, eliminating the cause of the Alert and/or waiting out prior Type D QDRs. The knob was provided by the SAG using SPAN methodology [1]. Documentation is found in Appendix E-18 and E-19.

Appendix I

Report Generator

```
                                                                        JOB 7469
1    //GOR6040A JOB (6040,GOR),'GROVER',CLASS=1,MSGCLASS=V               00030012
2    //RUNFTN EXEC FORTVCG                                               00010013
3    XXFORTVCG    PROC FVPGM=FORTVS,FVREGN=2400K,FVPDECK=NODECK,        00020000
     XX         FVPOLST=NOLIST,FVPOPT=O,FVTERM='SYSOUT=*',              00030003
     XX          FVLNSPC='3200,(25,6)',                                 00040000
     XX          GOF5DD='DDNAME=SYSIN',GOF6DD='SYSOUT=*',               00050007
     XX          GOF7DD='SYSOUT=B',GOREGN=2000K                         00060000
     ***                                                                00070009
4    XXFORT       EXEC PGM=&FVPGM,REGION=&FVREGN,                       00080012
     XX         PARM='&FVPDECK,&FVPOLST,OPT(&FVPOPT),NOTF'              00081011
5    XXSTEPLIB    DD   DSN=VSF.FORTVS,DISP=SHR                          00090000
6    XXSYSPRINT   DD   SYSOUT=*,DCB=BLKSIZE=3429                        00091002
7    XXSYSTERM    DD   &FVTERM                                          00100000
8    XXSYSPUNCH   DD   SYSOUT=B,DCB=BLKSIZE=3440                        00110014
9    XXSYSLIN     DD   DSN=&&LOADSET,DISP=(MOD,PASS),UNIT=WORKD,        00120002
     XX          SPACE=(&FVLNSPC),DCB=BLKSIZE=3200                      00040099
10   //FORT.SYSIN DD DSN=GOR.GROVER.FOR(RPT8),DISP=SHR                  00220010
11   XXGO    EXEC PGM=LOADER,REGION=&GOREGN,COND=(4,LT,FORT),          00221002
     XX         PARM='LET,NORES,EP=MAIN'                                00222002
12   XXSYSLIN     DD   DSN=&&LOADSET,DISP=(OLD,DELETE)                  00223002
13   XXSYSLOUT    DD   SYSOUT=*                                         00224011
14   XXSYSLIB     DD   DSN=VSF.VFORTLIB,DISP=SHR                        00225008
15   XX           DD   DSN=DGTS.FSUBLIB,DISP=SHR                        00230000
16   XXFT05F001   DD   &GOF5DD                                          00240000
17   XXFT06F001   DD   &GOF6DD                                          00250000
18   XXFT07F001   DD   &GOF7DD                                          00050099
19   //GO.FT12F001 DD DSN=GOR.GROVER.ATL.REPYN,DISP=SHR                 00060099
20   //GO.FT13F001 DD DSN=GOR.GROVER.SCORES.JUL87,DISP=SHR              00080012
21   //GO.FT06F001 DD SYSOUT=*                                          00090000
22   //SYSOUT DD SYSOUT=*                                               00100000
23   //SYSUDUMP DD SYSOUT=*                                             00110000
24   //SYSPRINT DD SYSOUT=*
```

```
C THIS IS THE DYNAMIC VERSION OF THE REPORT GENERATOR                    00010099
C THIS VERSION IS TO ADD PRIOR MONTH TO REPORT                           00020099
C THIS VERSION WEIGHTS RESIDENT FACILITIES BY SIZE WITHIN SECTIONS       00030099
C THIS VERSION WEIGHTS ALL LEVELS BY APPROX NUMBER OF OARS.              00040099
C AND WEIGHTS ALL LEVELS BY APPROX NUMBER OF OARS.                       00050099
C THIS VERSION GENERATES A PEER GROUP PERCENTILE FOR EACH OBSERVATION.   00060099
C ** THIS PROGRAM GENERATES A TAILORED REPORT                            00070099
C  DECLARE AND ARRAY VARIABLES                                           00080099
      CHARACTER ORGCD(500)*3,ORG*3,FSCM(500)*6,FLAG(500)*18,TYP(500)*1   00090099
      INTEGER MONTH(500),YEAR(500),DEGREE(500),YR,GRP(500)               00100099
      REAL EFF(11,500),REGCUM(11),SECCUM(11,3),BRACUM(11),DIVCUM(11)     00110099
      REAL A(628,2)                                                      00120099
C  INITIALIZE COUNTERS AND CUMULATORS                                    00130099
      CALL INIT(REGCUM,SECCUM,BRACUM,DIVCUM,BRAN,SECT,DIVN)              00140099
C  READ IN PEER SCORES                                                   00150099
      I=0                                                                00160099
      II=0                                                               00170099
    2 I=I+1                                                              00180099
      A(I,1)=80.0                                                        00190099
      A(I,2)=10.0                                                        00200099
      IF(II .GT. I) GOTO 2                                               00210099
      READ(13,14) II,A(I,1),A(I,2)                                       00211099
C     WRITE(6,14) II,A(I,1),A(I,2)                                       00220099
   14 FORMAT(17X,I3,25X,F8.4,3X,F8.4,68X)                                00220199
      IF(II .GT. I) THEN                                                 00220299
       A(II,1)=A(I,1)                                                    00220399
       A(II,2)=A(I,2)                                                    00220499
       DO 4 K=I,II-1                                                     00220599
        A(K,1)=80.0                                                      00220699
        A(K,2)=10.0                                                      00220799
    4  CONTINUE                                                          00220899
       I=II                                                              00220999
      ENDIF                                                              00221099
      IF(II .LT. 628) GOTO 2                                             00221199
C     DO 3 J=1,628                                                       00222099
C     WRITE(6,14) J,A(J,1),A(J,2)                                        00223099
C   3 CONTINUE                                                           00230099
C  READ FIRST RECORD                                                     00240099
      READ(12,15,END=99) ORGCD(1),FSCM(1),TYP(1),MONTH(1),YEAR(1),       00250099
     1GRP(1),DEGREE(1),FLAG(1),EFF(1,1),EFF(2,1),EFF(3,1),EFF(4,1),      00260099
     2EFF(5,1),EFF(6,1),EFF(7,1),EFF(8,1),EFF(9,1),EFF(10,1),EFF(11,1)   00270099
      DO 13 J=1,11                                                       00280099
       IF(TYP(1) .EQ. 'R') THEN                                          00281099
        IF(GRP(1) .EQ. 625) GRP(1)=991                                   00290099
         IRESCT=(MOD((GRP(1)+4),5)+1)**2                                 00300099
         SECCUM(J,1)=EFF(J,1)*REAL(IRESCT)                               00310099
         NRESCT=0                                                        00320099
        ELSE                                                             00330099
         SECCUM(J,2)=EFF(J,1)                                            00340099
         NRESCT=1                                                        00350099
         IRESCT=0                                                        00360099
       ENDIF                                                             00370099
   13 CONTINUE                                                           00380099
C  READ REMAINING RECORDS FOR SECTION                                    00390099
   11 LIN=0                                                              00400099
      DO 20 I=2,500                                                      00410099
      READ(12,15,END=99) ORGCD(I),FSCM(I),TYP(I),MONTH(I),YEAR(I),       00420099
     1GRP(I),DEGREE(I),FLAG(I),EFF(1,I),EFF(2,I),EFF(3,I),EFF(4,I),      00430099
     2EFF(5,I),EFF(6,I),EFF(7,I),EFF(8,I),EFF(9,I),EFF(10,I),EFF(11,I)   00440099
   15 FORMAT(A3,A6,A1,2I3,I4,I2,A18,7F7.1,F12.1,3F7.1)                   00450099
C  *** IF A NEW MONTH IS FOUND, UPDATE ALL SUMMARY STATS.                00460099
      IF(MONTH(I) .NE. MONTH(I-1)) THEN                                  00470099
  101   CALL HEADR(I,MONTH,YEAR,ORGCD)                                   00480099
        CALL WRITER(I,LIN,FSCM,GRP,DEGREE,FLAG,EFF,MONTH,YEAR,ORGCD,A)
```

```
      CALL SECTOT(I,SECCUM,IRESCT,NRESCT,EFF,ORGCD,FSCM,YEAR,MONTH,    00490099
     1GRP,DEGREE,FLAG,BRACUM,TYP,SECT,ORG,MON,YR)                      00500099
      CALL BRATOT(BRACUM,BRAN,SECT,DIVCUM,ORG)                         00510099
      CALL DIVTOT(DIVCUM,DIVN,BRAN,REGCUM,ORG)                         00520099
      CALL REGTOT(REGCUM,DIVN,MON,YR)                                  00530099
      IF(IENDR .EQ. 1) GOTO 100                                        00540099
      GO TO 11                                                         00550099
     ENDIF                                                             00560099
C  *** IF A NEW DIVISION IS FOUND, UPDATE ALL SUMMARY STATS.           00570099
     IF(ORGCD(I)(1:1) .NE. ORGCD(I-1)(1:1)) THEN                       00580099
      CALL HEADR(I,MONTH,YEAR,ORGCD)                                   00590099
      CALL WRITER(I,LIN,FSCM,GRP,DEGREE,FLAG,EFF,MONTH,YEAR,ORGCD,A)   00600099
      CALL SECTOT(I,SECCUM,IRESCT,NRESCT,EFF,ORGCD,FSCM,YEAR,MONTH,    00610099
     1GRP,DEGREE,FLAG,BRACUM,TYP,SECT,ORG,MON,YR)                      00620099
      CALL BRATOT(BRACUM,BRAN,SECT,DIVCUM,ORG)                         00630099
      CALL DIVTOT(DIVCUM,DIVN,BRAN,REGCUM,ORG)                         00640099
      GO TO 11                                                         00650099
     ENDIF                                                             00660099
C  *** IF A NEW BRANCH IS FOUND, UPDATE ALL SUMMARY STATS.             00670099
     IF(ORGCD(I)(1:2) .NE. ORGCD(I-1)(1:2)) THEN                       00680099
      CALL HEADR(I,MONTH,YEAR,ORGCD)                                   00690099
      CALL WRITER(I,LIN,FSCM,GRP,DEGREE,FLAG,EFF,MONTH,YEAR,ORGCD,A)   00700099
      CALL SECTOT(I,SECCUM,IRESCT,NRESCT,EFF,ORGCD,FSCM,YEAR,MONTH,    00710099
     1GRP,DEGREE,FLAG,BRACUM,TYP,SECT,ORG,MON,YR)                      00720099
      CALL BRATOT(BRACUM,BRAN,SECT,DIVCUM,ORG)                         00730099
      GO TO 11                                                         00740099
     ENDIF                                                             00750099
C  *** IF A NEW SECTION IS FOUND, UPDATE ALL SUMMARY STATS.            00760099
     IF(ORGCD(I)(1:3) .NE. ORGCD(I-1)(1:3)) THEN                       00770099
      CALL HEADR(I,MONTH,YEAR,ORGCD)                                   00780099
      CALL WRITER(I,LIN,FSCM,GRP,DEGREE,FLAG,EFF,MONTH,YEAR,ORGCD,A)   00790099
      CALL SECTOT(I,SECCUM,IRESCT,NRESCT,EFF,ORGCD,FSCM,YEAR,MONTH,    00800099
     1GRP,DEGREE,FLAG,BRACUM,TYP,SECT,ORG,MON,YR)                      00810099
      GO TO 11                                                         00820099
C  CURRENT RECORD IS IN THE SAME SECTION. UPDATE SECTION COUNTERS       00830099
     ELSE                                                              00840099
     IF(TYP(I) .EQ. 'R') THEN                                          00850099
      IF(GRP(I) .EQ. 625) GRP(I)=991                                   00851099
      IRESCT=IRESCT+(MOD((GRP(I)+4),5)+1)**2                           00860099
      DO 23 J=1,11                                                     00870099
       SECCUM(J,1)=SECCUM(J,1)+EFF(J,I)*(REAL(MOD((GRP(I)+4),5)+1))**200880099
                                                                       00890099
   23 CONTINUE                                                         00900099
      ELSE                                                             00910099
       NRESCT=NRESCT+1                                                 00920099
      DO 24 J=1,11                                                     00930099
       SECCUM(J,2)=SECCUM(J,2)+EFF(J,I)                                00940099
   24 CONTINUE                                                         00950099
     ENDIF                                                             00960099
     ENDIF                                                             00970099
   20 CONTINUE                                                         00980099
   98 WRITE(6,97)                                                      00990099
   97 FORMAT(1X,'TOO MANY FACILITIES ASSIGNED TO SECTION.ABNORMAL END') 01000099
C  NORMAL END . FINISH PROCESSING LAST ORGS AND END JOB.               01010099
   99 IENDR=1                                                          01020099
      GO TO 101                                                        01030099
  100 STOP                                                             01040099
      END                                                              01050099
      SUBROUTINE INIT(REGCUM,SECCUM,BRACUM,DIVCUM,BRAN,SECT,DIVN)      01060099
      REAL SECCUM(11,3),BRACUM(11),REGCUM(11),DIVCUM(11)               01070099
      DO 10 I=1,11                                                     01080099
       REGCUM(I)=0.0                                                   01090099
       SECCUM(I,1)=0.0                                                 01100099
       SECCUM(I,2)=0.0                                                 01110099
       SECCUM(I,3)=0.0                                                 01120099
       BRACUM(I)=0.0                                                   01130099
       DIVCUM(I)=0.0
```

```
 10 CONTINUE                                                            01140099
     BRAN=0.0                                                           01150099
     SECT=0.0                                                           01160099
     DIVN=0.0                                                           01170099
    RETURN                                                              01180099
    END                                                                 01190099
    SUBROUTINE HEADR(I,MONTH,YEAR,ORGCD)                                01200099
     INTEGER MONTH(500),YEAR(500)                                       01210099
     CHARACTER ORGCD(500)*3                                             01220099
 21   WRITE(6,17) MONTH(I-1),YEAR(I-1),ORGCD(I-1)                       01230099
 17   FORMAT('1',2I3,' EFFECTIVENESS REPORT FOR SECTION ',A3//' ',5X,   01240099
     1 'FSCM',4X,'GRP',7X,'RED FLAGS',5X,'PROGRAM',3X,'EPA',5X,'LRR',   01250099
     2 5X,'MRB',4X,'WVRS8',4X,'ECP',6X,'CA',5X,'MDR',3X,'PRODUCT',      01260099
     3 2X,'TOTAL',3X,'PRIOR',3X,'PEER'/' ',38X,'SCORE',27X,'DEVS',28X,  01270099
     4 'SCORE',3X,'SCORE',3X,'MONTH',2X,'RATING')                       01280099
     RETURN                                                             01290099
     END                                                                01300099
    SUBROUTINE WRITER(I,LIN,FSCM,GRP,DEGREE,FLAG,EFF,MONTH,YEAR,ORGCD,  01310099
     1A)                                                                01320099
     INTEGER DEGREE(500),MONTH(500),YEAR(500),GRP(500)                  01330099
     CHARACTER FLAG(500)*18,FSCM(500)*6,ORGCD(500)*3,RANKIT*1           01340099
     REAL EFF(11,500),A(628,2)                                          01350099
      DO 18 K=1,50                                                      01360099
       LIN=LIN+1                                                        01370099
       IF(LIN .GE. I) GOTO 22                                           01380099
       CALL PEER(K,GRP,EFF,A,RANKIT)                                    01390099
       IF(DEGREE(K)/2*2 .NE. DEGREE(K)) FSCM(K)(1:1)='*'                01400099
       IF(K .GT. 1) THEN                                                01410099
        IF((DEGREE(K-1) .LT. 3) .AND. (DEGREE(K) .GE. 3)) WRITE(6,15)   01420099
       ENDIF                                                            01430099
       WRITE(6,14) FSCM(K),GRP(K),FLAG(K),EFF(1,K),EFF(2,K),            01440099
     1EFF(3,K),EFF(4,K),EFF(5,K),EFF(6,K),EFF(7,K),EFF(8,K),EFF(9,K),   01450099
     2EFF(10,K),EFF(11,K),RANKIT                                        01460099
 14    FORMAT(4X,A6,3X,I4,1X,A18,2X,11(F5.1,3X),2X,A1)                  01470099
 15    FORMAT(' ')                                                      01480099
 18   CONTINUE                                                          01490099
    CALL HEADR(I,MONTH,YEAR,ORGCD)                                      01500099
 22 RETURN                                                              01510099
    END                                                                 01520099
    SUBROUTINE PEER(K,GRP,EFF,A,RANKIT)                                 01521099
    INTEGER GRP(500)                                                    01522099
    CHARACTER RANKIT*1                                                  01523099
    REAL EFF(11,500),A(628,2)                                           01524099
     IF (EFF(10,K) .GT. A(GRP(K),1)+A(GRP(K),2)) THEN                   01525099
      RANKIT='A'                                                        01526099
      GO TO 10                                                          01527099
      ENDIF                                                             01527199
     IF (EFF(10,K) .GT. A(GRP(K),1)+.5*A(GRP(K),2)) THEN                01528099
      RANKIT='B'                                                        01529099
      GO TO 10                                                          01529199
      ENDIF                                                             01529299
     IF (EFF(10,K) .GT. A(GRP(K),1)-.5*A(GRP(K),2)) THEN                01529399
      RANKIT='C'                                                        01529499
      GO TO 10                                                          01529599
      ENDIF                                                             01529699
     IF (EFF(10,K) .GT. A(GRP(K),1)-A(GRP(K),2)) THEN                   01529799
      RANKIT='D'                                                        01529899
      GO TO 10                                                          01529999
      ENDIF                                                             01530099
      RANKIT='F'                                                        01530199
 10 RETURN                                                              01530299
    END                                                                 01530399
    SUBROUTINE SECTOT(I,SECCUM,IRESCT,NRESCT,EFF,ORGCD,FSCM,YEAR,       01531099
     1MONTH,GRP,DEGREE,FLAG,BRACUM,TYP,SECT,ORG,MON,YR)                 01540099
     CHARACTER ORGCD(500)*3,FSCM(500)*6,FLAG(500)*18,TYP(500)*1,ORG*3   01550099
     INTEGER MONTH(500),YEAR(500),DEGREE(500),YR,GRP(500)               01560099
```

```
      REAL EFF(11,500),SECCUM(11,3),BRACUM(11),DIVCUM(11),REGCUM(11)      01570099
      REAL NMRTR, DNMNTR, RSWGT                                           01580099
      SECT=SECT+REAL(IRESCT)+REAL(NRESCT)/5.0                             01590099
      RSWGT=.2                                                            01600099
      DO 1 J=1,11                                                         01610099
       NMRTR=(SECCUM(J,1)+RSWGT*SECCUM(J,2))                              01620099
       DNMNTR=(REAL(IRESCT)+RSWGT*REAL(NRESCT)+.00001)                    01630099
       SECCUM(J,3)=NMRTR/DNMNTR                                           01640099
       SECCUM(J,1)=SECCUM(J,1)/(REAL(IRESCT)+.00001)                      01650099
       SECCUM(J,2)=SECCUM(J,2)/(REAL(NRESCT)+.00001)                      01660099
       BRACUM(J)=BRACUM(J)+SECCUM(J,3)*(REAL(IRESCT)+REAL(NRESCT)/5.0)    01670099
                                                                          01680099
     1 CONTINUE                                                           01690099
      WRITE(6,5) ORGCD(I-1),SECCUM(1,1),SECCUM(2,1),SECCUM(3,1),          01700099
     1SECCUM(4,1),SECCUM(5,1),SECCUM(6,1),SECCUM(7,1),SECCUM(8,1),        01710099
     2SECCUM(9,1),SECCUM(10,1),SECCUM(11,1)                               01720099
     5 FORMAT(' ',13('----------')/' ',' SUBTOTALS ','  ',2X,A3/' ',      01730099
      1'      RESIDENT',24X,11(F5.1,3X))                                  01740099
      WRITE(6,11) SECCUM(1,2),SECCUM(2,2),SECCUM(3,2),SECCUM(4,2),        01750099
     1SECCUM(5,2),SECCUM(6,2),SECCUM(7,2),SECCUM(8,2),SECCUM(9,2),        01760099
     2SECCUM(10,2),SECCUM(11,2)                                           01770099
    11 FORMAT(' ',5X,'NONRESIDENT',21X,11(F5.1,3X))                       01780099
      WRITE(6,12) SECCUM(1,3),SECCUM(2,3),SECCUM(3,3),SECCUM(4,3),        01790099
     1SECCUM(5,3),SECCUM(6,3),SECCUM(7,3),SECCUM(8,3),SECCUM(9,3),        01800099
     2SECCUM(10,3),SECCUM(11,3)                                           01810099
    12 FORMAT(' ',5X,'COMBINED    ',21X,11(F5.1,3X))                     01820099
      ORG=ORGCD(I-1)                                                      01830099
      MON=MONTH(I-1)                                                      01840099
      YR=YEAR(I-1)                                                        01850099
      ORGCD(1)=ORGCD(I)                                                   01860099
      FSCM(1)=FSCM(I)                                                     01870099
      YEAR(1)=YEAR(I)                                                     01880099
      MONTH(1)=MONTH(I)                                                   01890099
      TYP(1)=TYP(I)                                                       01900099
      GRP(1)=GRP(I)                                                       01910099
      DEGREE(1)=DEGREE(I)                                                 01920099
      FLAG(1)=FLAG(I)                                                     01930099
      DO 20 J=1,11                                                        01940099
       EFF(J,1)=EFF(J,I)                                                  01950099
       IF (TYP(1) .EQ. 'R') THEN                                          01951099
         IF(GRP(1) .EQ. 625) GRP(1)=991                                   01960099
         IRESCT=(MOD((GRP(1)+4),5)+1)**2                                  01970099
         NRESCT=0                                                         01980099
         SECCUM(J,1)=EFF(J,1)*REAL(IRESCT)                                01990099
         SECCUM(J,2)=0                                                    02000099
       ELSE                                                               02010099
         IRESCT=0                                                         02020099
         NRESCT=1                                                         02030099
         SECCUM(J,1)=0                                                    02040099
         SECCUM(J,2)=EFF(J,1)                                             02050099
       ENDIF                                                              02060099
    20 CONTINUE                                                           02070099
      RETURN                                                              02080099
      END                                                                 02090099
      SUBROUTINE BRATOT(BRACUM,BRAN,SECT,DIVCUM,ORG)                      02100099
      REAL BRACUM(11),DIVCUM(11)                                          02110099
      CHARACTER ORG*3                                                     02120099
      DO 10 K=1,11                                                        02130099
       DIVCUM(K)=DIVCUM(K)+BRACUM(K)                                      02140099
       BRACUM(K)=BRACUM(K)/SECT                                           02150099
    10 CONTINUE                                                           02160099
      WRITE(6,15) ORG(1:2)                                                02170099
    15 FORMAT(' ',13('----------')/' ',' SUBTOT ','  ',2X,A2)             02180099
       WRITE(6,40) BRACUM(1),BRACUM(2),BRACUM(3),BRACUM(4),BRACUM(5),     02190099
     1BRACUM(6),BRACUM(7),BRACUM(8),BRACUM(9),BRACUM(10),BRACUM(11)       02200099
    40 FORMAT('+',37X,11(F5.1,3X))                                        02210099
      DO 16 J=1,11
```

```
      BRACUM(J)=0.0                                                     02220099
   16 CONTINUE                                                          02230099
      BRAN=BRAN+SECT                                                    02240099
      SECT=0.0                                                          02250099
      RETURN                                                            02260099
      END                                                               02270099
      SUBROUTINE DIVTOT(DIVCUM,DIVN,BRAN,REGCUM,ORG)                    02280099
      REAL DIVCUM(11),REGCUM(11)                                        02290099
      CHARACTER ORG*3                                                   02300099
      DO 10 K=1,11                                                      02310099
       REGCUM(K)=REGCUM(K)+DIVCUM(K)                                    02320099
       DIVCUM(K)=DIVCUM(K)/BRAN                                         02330099
   10 CONTINUE                                                          02340099
      WRITE(6,35) ORG(1:1)                                              02350099
   35 FORMAT(' ',13('----------')/' ',' SUBTOT ',' ',2X,A1)             02360099
      WRITE(6,40) DIVCUM(1),DIVCUM(2),DIVCUM(3),DIVCUM(4),DIVCUM(5),    02370099
     1DIVCUM(6),DIVCUM(7),DIVCUM(8),DIVCUM(9),DIVCUM(10),DIVCUM(11)     02380099
   40 FORMAT('+',37X,11(F5.1,3X))                                       02390099
      DO 36 J=1,11                                                      02400099
       DIVCUM(J)=0.0                                                    02410099
   36 CONTINUE                                                          02420099
      DIVN=DIVN+BRAN                                                    02430099
      BRAN=0.0                                                          02440099
      RETURN                                                            02450099
      END                                                               02460099
      SUBROUTINE REGTOT(REGCUM,DIVN,MON,YR)                             02470099
      REAL REGCUM(11)                                                   02480099
      INTEGER YR                                                        02490099
      DO 10 K=1,11                                                      02500099
       REGCUM(K)=REGCUM(K)/DIVN                                         02510099
   10 CONTINUE                                                          02520099
      WRITE(6,45) MON,YR                                                02530099
   45 FORMAT(' ',13('----------')/' ',' REGION SUMMARY ',' ',2X,2I3)     02540099
      WRITE(6,40) REGCUM(1),REGCUM(2),REGCUM(3),REGCUM(4),REGCUM(5),    02550099
     1REGCUM(6),REGCUM(7),REGCUM(8),REGCUM(9),REGCUM(10),REGCUM(11)     02560099
   40 FORMAT('+',37X,11(F5.1,3X))                                       02570099
      DO 46 J=1,11                                                      02580099
       REGCUM(J)=0.0                                                    02590099
   46 CONTINUE                                                          02600099
      DIVN=0.0                                                          02610099
      RETURN                                                            02620099
      END                                                               02630099
```

Appendix J

References

1. Hwang, Ching-Lai and Lin, Ming-Jeng, Group Decision Making Under Multiple Criteria, Springer-Verlag, Berlin Heidleberg, 1987.

2. Hwang, Ching-Lai and Yoon, Kwangsun, Multiple Attribute Decision Making, Springer-Verlag, Berline Heidleberg and New York, 1981.

3. Freund, John E., Modern Elementary Statistics, Prentice-Hall, Third Edition, Englewood Cliffs, NJ, 1967.